FULL-POTENTIAL LMTO PROGRAMS "NMT"
USER's MANUAL

S.Yu.SAVRASOV
Max-Planck Institute fuer Festkoerperforschung, D-70569 Stuttgart, Germany.
Department of Physics and Astronomy, Rutgers University, Piscataway, NJ 08854.

October 13, 2000

Contents

1 INTRODUCTION **2**

2 INSTALLATION **4**

3 MAIN CONTROL FILE: INIFILE **6**
3.1 Control Parameters . . . 8
3.2 Exchange-correlation functional . . . 10
3.3 Iterative Procedures Limits and Accuracies . . . 10
3.4 Atomic Data . . . 11
3.5 Output Control Parameters: . . . 18
3.6 Other Data for NMT-pack . . . 20

4 STRUCTURE CONTROL FILE: STRFILE **23**
4.1 Main Parameters . . . 24
4.2 Primitive Translations . . . 25
4.3 Basis in original Unit-Cell . . . 25
4.4 Basis in distorted Unit-Cell . . . 25
4.5 Strain matrix . . . 26
4.6 Point Group Description . . . 26
4.7 Primitive Translations for BZ . . . 27
4.8 High-symmetry direction settings . . . 27
4.9 Settings for PLOBANDS program . . . 27

5 INPUT CHARGE DENSITY FILE: SCFFILE **28**

6 RUNNING THE PROGRAM **29**

7 OUTPUT MESSAGE FILE: OUTFILE **30**
7.1 Reading input data . . . 30
7.2 Preparing structure constants . . . 32
7.3 Computing sphere radii . . . 33
7.4 Finding full potential . . . 34
7.5 Calculating energy bands . . . 35
7.6 Integrating over Brillouin zone . . . 35
7.7 Constructing the charge density . . . 36
7.8 Renormalizing core levels . . . 36
7.9 Evaluating total energy . . . 36
7.10 Evaluating forces . . . 37
7.11 Mixing charge densities . . . 37

8 PARAMETER FILE: PARAM.DAT **39**
8.1 Main parameters . . . 39
8.2 Parameters for plane waves . . . 40
8.3 Parameters for cells . . . 41
8.4 Frequently used settings . . . 41
8.5 Estimation of the needed core memory . . . 41

9 ERROR MESSAGES **42**
9.1 Errors connected with PARAM.DAT . . . 42
9.2 Errors connected with input . . . 42
9.3 Errors connected with iterative procedures . . . 42
9.4 Other errors . . . 42

10 VERSIONS DIFFERENCES **43**

11 USING LOCAL LIBRARY **44**
11.1 Program CELL . . . 44
11.2 Program GRID . . . 44
11.3 Program SCF1 . . . 45
11.4 Program SCFM . . . 45

12 USING NMTLIB LIBRARY **46**
12.1 Program ATOM . . . 46
12.2 Program CHIQ . . . 47
12.3 Program MESH . . . 47
12.4 Program MLSQ . . . 47
12.5 Program PLOT . . . 47
12.6 Program QPNT . . . 48
12.7 Program SURF . . . 48

13 ADDITIONAL INPUT: HUBFILE **49**
13.1 General Settings 53
13.2 Description of Correlated States 54
13.3 Table of Occupation Numbers 54

14 ADDITIONAL INPUT: HOPFILE **55**
14.1 General Settings 55
14.2 Description of Hopping Integrals 55

15 ADDITIONAL INPUT: KOVFILE **57**

16 Acknowledgents **63**

17 COPYRIGHT **63**

1 INTRODUCTION

The full-potential linear-muffin-tin-orbital (Ref. [1]) (FP-LMTO) programs described here are designed to perform band structure, total energy and force calculations within the methods of density functional theory (DFT) (Refs. [2, 3, 4]). Main features include:

- (i) Local (spin) density approximation (LSDA) avaiable in many parametrizations together with the gradient corrected density functionals (GGA91 & GGA96).
- (ii) Multiple-κ LMTO basis sets and many panel technique.
- (iii) Total energy and force calculations for determining the equillibrium structure and phonons.
- (iv) LDA+U method for strongly correlated systems.
- (v) Spin-orbit coupling for heavy elements.

At the moment, three versions of the full-potential code are available which use essentially the same input files. (Some data in the input file are ignored for one or another version). These versions are:

- NMTASA: overlapping atomic spheres, potential is non-spherical inside the spheres, no interstitial region. Fast and dirty: provides reasonably good bands, but is not sufficiently acurate for phonons and distortions. (*Version 1.60* of this package or the later ones should match to this manual.)
- NMTCEL: unit cell is divided into polyhedra, potential is expanded in the spherical harmonics inside the polyhedra. Has intermediate computation speed and provides reasonable accuracy for many phonon/elastic problems provided that the structure is reasonably close packed. The description of this method has been published in Ref. [5]. (*Version 3.80* of this package or the later ones should match to this manual.)
- NMTPLW (plane waves): non-overlapping muffin-tin (MT) spheres, potential is expanded in spherical harmonics inside the spheres and Fourier transformed in the interstitial region. Provides the best accuracy at the price of increasing the computation time. A short description of this method can be found in Ref. [6]. (*Version 2.50* of this package or the later ones should match to this manual.)

Main parts which run independently (for each version) are:

- Module which makes a starting charge density file for the band structure calculation from the charge densities of constituent atoms.
- Package for the self-consistent FP calculation (or, optionally, energy bands along directions).
- Additional programs for plotting the bands, densities of states, charge densities, etc.

All programs can be obtained by contacting S. Savrasov using the e-mail address: savrasov@and.mpi-stuttgart.mpg.de

About the notations in this document:

- all file names like **nio.ini, main.exe**, are boldfaced.
- all directory names like */nmtplw/lib/scfm/* are italicised.
- symbol "*" in the directory name *nmt** refers to either *asa,cel* or *plw*. In other cases it has the standard meaning.
- capitalized names like INIFILE, STRFILE are made to shorten references to the MAIN INPUT CONTROL FILE (for INIFILE), STRUCTURE CONTROL FILE (for STRFILE), etc.

I apologize if the description of some parameters is short and unclear. Any suggestions to improve this manual are welcomed.

2 INSTALLATION

In this section the directories which are used for running the programs and storing input/output data are described. All NMT* programs have the same directory organization containing (see Figure 1)

- */nmt*/run/* - directory containing the source code ***.f** (text of the program written on FORTRAN77), object files ***.o** and executable file usually named as **main.exe**.
- */nmt*/dat/* - directory with the input/output data files. Usually, many subdirectories are created here according to the element (compound) name to be calculated.
- */nmt*/lib/* - directory containing the programs for helping to construct input data files and understand output information. For example, subdirectory */nmt*/lib/scfm/* contains the program which creates input charge density distrubtion. Subdirrectory */nmt*/lib/cell/* containes the program which helps to find the muffin-tin sphere radii. Subdirectory */nmt*/lib/grid/* creates two dimensional grid for plotting the self-consistent charge densities. Each subdirectory containes source data files ***.f**, object files ***.o** and the executable file usually named **make.exe**.

In contrast to the local libraries which exist inside */nmtasa/,/nmtcel/,/nmtplw/*, there also exists a global library directory */nmtlib/*. The latter containes general purpose application programs. The programs understand output data files of all NMT* codes. These programs are

- */nmtlib/plot/* - programs to plot energy bands and densities of states.
- */nmtlib/chiq/* - program to calculate real/imaginary part of the susceptibility in the constant-matrix-element approximation.
- */nmtlib/mlsq/* - least-squares-fit program and sample input data to fit total energies and forces in order to find equillibrium configuration, bulk modulus, phonon frequencies, etc.
- */nmtlib/mesh/* - program to generate whole 3D grid of bands from the knowledge of the data at the irreducible part.
- */nmtlib/surf/* - sample input data to plot 3D graphs like Fermi surfaces by Data Explorer.
- */nmtlib/qpnt/* - program which generates irreducible set of **k**-points. according to the input division. Helpful to understand how many **k**-points is produced by certain division of the Brillouin zone. Also helpful for setting up perturbation wavevectors in linear response calculations of susceptibilities, phonons, etc.
- */nmtlib/atom/* - standard program after Liberman et al. to solve the free atom problem. The corresponding input data files are stored in the */atomdat/* directory. In fact, all free-atom charge densities have been calculated already for each element and also stored in this directory. This directory is only used as a library to create input charge density file.

All programs and data files are tared, gzipped, and uuencoded into 5 self-extractory files named as **nmtasa.dec**, **nmtcel.dec**, **nmtplw.dec**, **nmtlib.dec**, and **atomdat.dec**. To unpack them use the following commands:

1. **uudecode nmtasa.dec**

2. **gunzip nmtasa.tar.gz**

3. **tar -x -f nmtasa.tar**

Repeat these steps for **nmtcel.dec**, **nmtplw.dec**, **nmtlib.dec**, and **atomdat.dec**. The directory trees will be created automatically as illustrated in Figure 1.

To be able to run any of the full-potential LMTO programs it is necessary to compile the source data files. A few comments must be said here. First, the maximum size of every array (such as maximum number of atoms, $l_{\max}$, etc) in the program is declared using the fortran PARAMETER statement. These statements are contained in the file PARAM.DAT. (See Section PARAMETER FILE: PARAM.DAT for the detailed description) The file PARAM.DAT is included into every source code during the compilation time by INCLUDE statement. The second comment concerns a scratch file storage. To minimize core memory, some data during the run are temporary stored in the scratch files. To be able to do this, a scratch directory must exist on any particular node where execution of the program is performed.

To create executable file, go to the directory */nmt*/run/*.

1. Edit PARAM.DAT and install the necessary size of arrays. Sample PARAM.DAT file is provided.

2. Edit the file **setup.f** and specify the path to the scratch directory. Also check that other items match your computer settings.

3. Edit the file **timel.f** and specify the call to the *system* subroutine to learn CPU time.

4. Compile all programs, link them to get exectable file **main.exe**. Under UNIX, using AIX XL Fortran Compiler this looks like: **xlf -cOw *.f** to compile only, with optimization, and suppressing all warning messages. The command **xlf -cCg *.f** will compile only, suppress optimizationm and provide debugging information. To link, use the command **xlf *.o -o main.exe**. To create a load map use the command **xlf *.o -o main.exe -bloadmap:map**. At the end of the map file a total ammount of the core memory allocated by the program is printed out.

All NMT* programs have three main input files:

- INIFILE - main control data file which is the same for all NMT* codes.
- SCFFILE - input charge density file. Due to a different representation of the charge density used in ASA,CEL and RUN, this file is unique for every particular NMT code.
- STRFILE - structure data file describing the crystalline lattice.

Examples of INIFILEs and STRFILES are provided in */nmt*/dat/*.

3 MAIN CONTROL FILE: INIFILE

The main control file of the full-potential package has extension **.ini**. Below an example of band structure calculation for NiO will be considered. The INIFILE for NiO can be named as **nio.ini** or more shortly **ini**. It will be located in */nmt*/dat/nio/***nio.ini**. An example of this file is given below:

```
     *** Band Structure Calculation of NiO  ***
-----------------------------------------------------------------
Control Parameters:
 3                              - lift
 1                              - lmto
 0                              - nlrt
 2                              - lrwf
 0                              - npfr
Exchange-Correlation:
 3                              - 1,2,3 :by VBarth-H;Gunn-L;Jan-W
Iterative Procedure Limits and Accuracy:
 50,0.2,0.2,1.D-9,+2 1,0.3,0.3  - niter,mix,mag,eps,lbr,ibr,mixb,mixh
Atomic Data:
 4,3,2,1                        - natom,nsort,nspin,norbs
 8.0510,1.0                     - lattice parameter,v/v0
 1,2,3,3                        - is(iatom)
 3,(-0.6,0.0),(0.4,0.03),(1.4,0.03)     - nkap,ekap in valence states
for Ni1                         ----------------------------
 28.D0 12.D0,1,1,0,0.50,58.7    - z zcor,lr,icor,ispl,split,mass
 2.100,3.500,1.1,0.5            - mt-sphere,rou-sphere,weight,rloc
 6 2 6                          - lmax-t,lmax-b,lmax-v
Valence states are:
 s p d f                        - states for E=-0.6 Ry
 4 4 3 4                        - main quantum numbers
 0 0 0 0                        - basis set
 0 0 0 0                        - choice of Eny
-0.5,-0.5,-0.5,-0.5             - Eny
-1.0,-2.0,-3.0,-4.0             - Dny
 s p d f                        - states for E=0.4 Ry
 4 4 3 4                        - main quantum numbers
 1 1 1 0                        - basis set
 3 3 1 0                        - choice of Eny
 0.5, 0.5, 0.5, 0.5             - Eny
-1.0,-2.0,-3.0,-4.0             - Dny
 s p d f                        - states for E=1.4 Ry
 4 4 3 4                        - main quantum numbers
 1 1 1 0                        - basis set
 3 3 1 0                        - choice of Eny
 0.5, 0.5, 0.5, 0.5             - Eny
-1.0,-2.0,-3.0,-4.0             - Dny
semicore states are:
 1                              - # of semicore states
 3 1 (-4.0,0.0)                 - n,l, energy for 3p state
for Ni2                         ----------------------------
 28.D0 12.D0,1,1,0,-0.5,58.7    - z zcor,lr,icor,ispl,split,mass
 2.100,3.500,1.1,0.5            - mt-sphere,rou-sphere,weight,rloc
```

```
 6 2 6                          - lmax-t,lmax-b,lmax-v
valence states are:
 s p d f                        - states for E=-0.6 Ry
 4 4 3 4                        - main quantum numbers
 0 0 0 0                        - basis set
 0 0 0 0                        - choice of Eny
-0.5,-0.5,-0.5,-0.5             - Eny
-1.0,-2.0,-3.0,-4.0             - Dny
 s p d f                        - states for E=0.4 Ry
 4 4 3 4                        - main quantum numbers
 1 1 1 0                        - basis set
 3 3 1 0                        - choice of Eny
 0.5, 0.5, 0.5, 0.5             - Eny
-1.0,-2.0,-3.0,-4.0             - Dny
 s p d f                        - states for E=1.4 Ry
 4 4 3 4                        - main quantum numbers
 1 1 1 0                        - basis set
 3 3 1 0                        - choice of Eny
 0.5, 0.5, 0.5, 0.5             - Eny
-1.0,-2.0,-3.0,-4.0             - Dny
semicore states are:
 1                              - # of semicore states
 3 1 (-4.0,0.0)                 - n,l, energy for 3p state
for O                           ------------------------------
 8.D0 2.D0,1,1,0,0.D0,16.0      - z zcor,lr,icor,ispl,split,mass
 1.910,3.350,1.0                - mt-sphere,rounded sphere,weight
 6 1 6                          - lmax-t,lmax-b,lmax-v
valence states are:
 s p d f                        - states for E=-0.6 Ry
 2 2 3 4                        - main quantum numbers
 1 0 0 0                        - basis set
 2 0 0 0                        - choice of Eny
-0.5,-0.5,-0.5,-0.5             - Eny
-1.0,-2.0,-3.0,-4.0             - Dny
 s p d f                        - states for E=0.4 Ry
 2 2 3 4                        - main quantum numbers
 0 1 0 0                        - basis set
 0 3 0 0                        - choice of Eny
 0.5,0.5,0.5,0.5                - Eny
-1.0,-2.0,-3.0,-4.0             - Dny
 s p d f                        - states for E=1.4 Ry
 2 2 3 4                        - main quantum numbers
 0 1 0 0                        - basis set
 0 3 0 0                        - choice of Eny
 0.5,0.5,0.5,0.5                - Eny
-1.0,-2.0,-3.0,-4.0             - Dny
semicore states are:
 0                              - # of semicore states
Output Control Parameters:
 2 'nio.con'                    - icon,<confile>
 0 'nio.srf'                    - isrf,<srffile>
```

```
 0 'nio.psi'                    - ipsi,<psifile>
 0 'nio.plw'                    - itmp,<tmpfile>
 0 'nio.bnd'                    - ibnd,<bndfile>
 0 'nio.pot'                    - ipot,<potfile>
 0 'nio.ptn'                    - iptn,<ptnfile>
 0 'nio.dos'                    - idos,<dosfile>
 2 'nio.scf'                    - iscf,<scffile>
 2 'nio.out'                    - iout,<outfile>
Other Data for NMT-Pack:
-1,4,0.0                        - nff,nef,pole
 8,8,8,2                        - n1,n2,n3,nc
 32,32,32,0.02,0.04,1,5         - nfft1,nfft2,nfft3,epsR,epsG,kbz,rbz
Additional input files:
 2 'nio.hub'                    - ihub,<hubfile>
 0 'nio.hop'                    - ihop,<hopfile>
 0 'nio.opt'                    - iopt,<optfile>
 0 'nio.enr'                    - ienr,<enrfile>
 0 'nio.pnt'                    - ipnt,<pntfile>
```

The meaning of the entries is explained below:

3.1 Control Parameters

```
*** Band Structure Calculation of NiO  ***
-----------------------------------------------------------------
Control Parameters:
 3                              - lift
 1                              - lmto
 0                              - nlrt
 2                              - lrwf
 0                              - npfr
Exchange-Correlation:
 3                              - 1,2,3 :by VBarth-H;Gunn-L;Jan-W, etc.
...
```

- *lift*
  - 1: Calculate structural data files, then stop. This includes structure constants of the LMTO method used by all NMT* codes, and/or cell surface constants for NMTCEL program. Setting *lift*=1 is useful if the structural information is necessary for a number of jobs which will be executed spontaneoulsly.
  - 2: Calculate structural data files and make one calculation of the energy bands. This is useful if energy bands are necessary to calculate and store in one of the output files. No BZ integration is performed and the new charge density is not constructed.
  - 3: The whole self-consistent looping is switched on. This normally includes ppreparation of structural data files, calculation of energy bands, integration over BZ, charge density construction, evaluation of total energy, forces and preparation of the new charge density for the next iteration.
- *lmto*

- 1: Unscreened (original) LMTO. All NMT* programs use unscreened long-range LMTO representation as originally formulated in Ref.1
- 2: Screened (tight-binding) LMTO is under construction. It is avaiable in its simplest version in the program FTBASA (not automatically distributed but can be obtained upon request). Other versions FTBCEL and FTBPLW are not currently avaiable.

- *ncou* is the parameter which is only used by the NMTCEL program. *ncou*=1 switches on the Coulomb corrections to the full potential. This frequently improves the accuracy of the calculation in case of open lattices, and supercells by solving more accurately the Poisson equation in the interstitial region. The parameter has no effect on NMTASA and NMTPLW codes.

- *lrwf* substitutes previous parameter *nint*. The latter was used to control the calculation of interstitial overlap integrals. All current versions of the programs calculate overlap matrices using energy derivative of the structure constants. *nint*=2 is set in the program **setup.f** as a default value.

  The new parameter *lrwf* is used to control adjustment of the radial wave functions to the spherical part of the potential. Normally, the radial Schrödinger's equation is solved with the spherical potential at the current iteration. In case of spin polarized calculation, the equation is solved for both spin-up and spin-down potential and, therefore, radial wave functions have a spin dependence. There are however special cases when it becomes useful to solve radial Schrödinger's equation not with the spin dependent potential, but with the average potential (Vup+Vdn)/2. This eliminates explicit dependence of the radial wave functions from the spin index. It is necessary, for example, when calculating susceptibility functions using linear response theory. Another option is provided to froze radial wave functions for one particular spherical potential and do not recalculate them at every iteration of the self-consistency. If radial wave functions are frozen, then the calcuation of forces is exact in the sence that the calcuated force is exact derivative of the LMTO expression of the total energy without any further assumptions.

  The parameter *lrwf* can take one of the following values:

  - 2: Adjust radial wave functions to spin dependent potential. This is what is usually done and must be used in most cases.
  - 1: Adjsut radial wave functions to spin average part of the potential. This eliminates the dependence of radial wave functions from the spin index. It is necessary, when calculating dynamical susceptibility functions using linear response theory and the program MAGPLW (currently not avaiable but coming soon).
  - 0: Do not recalculate radial wave functions. The spherical potential (for which the radial wave functions will be constructed) must be stored in the POTFILE (see below). This feature should bring complete correspondence between calculated total energies and forces if there is a trouble that the calculated forces are inaccurate. In fact, it is useful for debugging purposes only, generally, the force formulae programmed are sufficiently accurate.

- *npfr*

  - 0: no accurate atomic force calculation. The output will only contain the Hellmann-Feuynman forces which are normally not accurate at all when using the LMTO method due to the large incomplete basis set (or Pulay) corrections.

– 1: atomic forces including both the Hellmann-Feynmann and Pulay contributions will be evaluated. The accuracy of the forces due to unself-consistency of the charge density can be controlled. Since evaluation of the Pulay forces is computationally demanding, switching this option is recommended after the self-consistency is reached.

– 2: more fast option for force calculations but with shorter output. The accuracy of forces due to unself-consistency cannot be controlled. In fact forces are accurate only with PLW package. They are absent in ASA code. See also file */nmtplw/run/***forces.f** for more comments.

3.2 Exchange-correlation functional

```
......
 0                              - npfr
Exchange-Correlation:
 3                              - 1,2,3 :by VBarth-H;Gunn-L;Jan-W, etc.
Iterative Procedure Limits and Accuracy:
 20,0.2,0.2,1.D-9,+2 0,0.3,0.3  - niter,mix,mag,eps,lbr,ibr,mixb,mixh
......
```

- 0: not inculded
- 1: von Barth–Hedin
- 2: Gunnarsson–Ludqvist
- 3: Janak–Moruzzi–Williams
- 4: Vosko–Wilk–Nussair
- 5: Perdew–Wang (local part of GGA-1991, similar to Vosko–Wilk–Nussair)
- 6: Gaspar–Kohn–Sham (no correlation)

To switch on generalized gradient approximation, add 10 to the key above for GGA91 of Perdew and Wang. Most recent GGA96 after Perdew et.al is switched on by adding 20. Therefore, valid keys are: Note that in PARAM.DAT file there is a parameter NGGAMAX which must be set to 3 for LDA+GGA calculations.

- 0,1,3,4,5,6 for LSDA only.
- 11,12,13,14,15,16 for LSDA+GGA91
- 21,22,23,24,25,26 for LSDA+GGA96

3.3 Iterative Procedures Limits and Accuracies

```
.....
Exchange-Correlation:
 3                              - 1,2,3 :by VBarth-H;Gunn-L;Jan-W, etc.
Iterative Procedure Limits and Accuracy:
```

```
 20,0.2,0.2,1.D-9,+2 0,0.3,0.3  - niter,mix,mag,eps,lbr,ibr,mixb,mixh
Atomic Data:
 4,3,2,1                        - natom,nsort,nspin,norbs
......
```

- *niter*: max. number of iterations in this run
- *mix*: starting mixing of the charge density in linear mixing scheme. During the iterations towards self-consistency the mixing will be optimally adjasted according to the Pratt scheme. This parameter is ignored if Broyden mixing (see below) is switched on.
- *mag*: starting mixing for the magnetization in linear mixing scheme. During the iterations towards self-consistency the magnetization mixing will remain constant and will NOT be adjusted. The parameter has no effect for non-spin-polarized calculations or if Broyden mixing (see below) is switched on.
- *eps*: total energy convergency criterion. Typically 10^{-6} Ry. The program will stop if the total energy difference between two consequent iterations is less then *eps*.
- *lbr*: switches on the Broyden mixing. If -1 then Broyden is OFF, if 0 then Broyden is ON for l=0 component of $\rho(\mathbf{r})$, if +1,+2 then Broyden is on for l.le.*lbr* component of rho(r) Recommended value is +1 since it does not take much disk space [because Broyden saves $\rho(\mathbf{r})$ for all previous iterations.] Broyden restarts every time after NTERMAX iterations. Parameter NTERMAX is in PARAM.DAT. Usually NTERMAX=15.
- *ibr* starts Broyden after I=*ibr* iterations. If *ibr*=0 then start immidiately. If *ibr*>0 then first I=*ibr* iterations will be done with the linear mixing scheme where the mixing parameters *mix* and *mag* are specified above.
- *mixb*: this is initial guess for Jacobian which is closely related to mixing parameter *mix* in the linear mixing scheme. It was found that *mixb* cannot be small and it is usually of the order 0.3-0.4
- *mixh*: this is linear minxing parameter for higher l-components (l>*lbr*) of the charge density. Since it is assumed that these components do not influence much the self-consistence loop, they are mixed within linear mixing scheme and do not stored for all previous iterations.

3.4 Atomic Data

```
.........
Atomic Data:
 4,3,2,1                        - natom,nsort,nspin,norbs
 8.0510,1.00                    - lattice parameter,v/v0
 1,2,3,3                        - is(iatom)
 3,(-0.6,0.0),(0.4,0.03),(1.4,0.03)     - nkap,ekap in valence states
.........
```

- *natom*: total number of atoms per unit cell
- *nsort*: the number of non-equivalent atoms

- *nspin*:
  - 1 for non-spin-polarized calculations
  - 2 for spin-polarized calculations
- *norbs*:
  - 1 without spin-orbit coupling
  - 2 incuding effects of spin-orbit coupling. This works only in combination with the switch *nspin*=2. In this case the size of the hamiltonian is doubled. Note that in the PARAM.DAT file PARAMETER NDIMMAX=..., (the size of the hamiltonian) need not to be increased. Inside the program the H matrix is dimensioned as HMAT(NDIMMAX*NSPINMAX, NDIMMAX*NSPINMAX) Note also that you cannot specify *nspin*=1, and *norbs*=2, i.e. spin-orbit coupling and NO spin-polarization. If spin-orbit coupling is ON, then spin-polarization is always assumed. In case non-magnetic calculation is required like Pb, for example do not specify an initial splitting of the potential (see parameter *split* below). If, after self-consistency is reached for non-magnetic spin-orbit coupled calculation, the spin-polarization is necessary, specify some splitting, and set *ispl*=2 (see also below). Do not forget to set *ispl*=0 after one run, since *ispl*=2 will always split the potential at the begining of every run.

    Notes to orbital magnetizm: spin-polarized spin-orbit coupled calculation makes non-zero average orbital moment. The program calculates orbital contribution to the magnetic moment and prints it out. However no contribution to the potential arizes from the orbital moment in LSDA! Therefore, the spin densities remain unchanged inside the program. In all places where the magnetic moment is calculated and printed out, it is SPIN magnetic moment, WITHOUT orbital contribution. The orbital contribution is printed out separately and must be added to the spin moment in order to obtain the total magnetic moment.

    Notes to the group symmetry: since spin-orbit coupling operator lowers the symmetry of crystal group, do not wonder if after switching SO-coupling, the crystal group will contain only 8 operations instead of 48 in the cubic case.
- *lattice parameter* in atomic units
- v/v_0: uniform compression
- *is(iatom)*: for each atom from 1 to *natom*, gives the sort of this atom. (The sequence of atoms as in the STRFILE, see below).
- *nkap*: number of different tail energies $E = \kappa^2$ in the valence band.
- *ekap*: energies $E = \kappa^2$ (complex numbers). If $\mathrm{Re}(\kappa^2)>0$ then $\mathrm{Im}(\kappa^2)$ must be non-zero and close to 0.03 Ry (to avoid singularities in the Ewald summations).

Then, sets of data for all non-equivalent sorts follow. Each set includes:

```
......
for Ni1                          ----------------------------
 28.D0 12.D0,1,1,0,0.50,58.7     - z zcor,lr,icor,ispl,split,mass
 2.100,3.500,1.1,0.5             - mt-sphere,rou-sphere,weight,rloc
 6 2 6                           - lmax-t,lmax-b,lmax-v
......
```

- *for Ni1*: title for every atom. Note that this character string (maximum 10 letters) will be read and widely used in the output file. Therefore it is recommended to use the format "for EL"
- *z*: atomic number, (if empty sphere, specify *z=0*)
- *zcor*: deep-core charge, i.e. those atomic states which will be treated as levels, not as bands. The program can treat three kinds of states: (i) Valence states, i.e. those which form the bands. These states form main valence panel and will be found by diagonalizing LMTO hamiltonian. (ii) Semicore states, i.e. states which have small but negligible band width. These states have small hybridization with the valence states, and therefore, are treated in separate energy panels. They also found by diagonalizing LMTO hamiltonian corresponding to each of the semicore state. (iii) Deep-core states, which are found by solving Dirac's equation for free atom with the potential taken as the spherical part of the crystalline potential until the MT-sphere and zero outside it. A simple rule to sort out the states over these three sets is the following: if the free-atom state (see corresponding **rat.<element>** file in the directory */atomdat/*) has an energy approximately above -2 Ry from the vacuum zero (or above -1 Ry from the MT-zero, but note that the latter is not known until the band structure calculation is performed), then this state should be treated as valence state and described in the main valence panel. If the state has an energy lying between -6 and -2 Ry from the vacuum zero, this state should be treated as a semicore state and must be described in one of the semicore panel. All other low–lying states can be treated as levels.
- *lr*:
  - 0 for non-relativistic calculations,
  - 1 for scalar-relativistic valence states
- *icor*:
  - 0: frozen deep core
  - 1: soft deep core (always fully relativistic)
- *ispl,split*:
  - *ispl*=0: without permanent splitting of the spin-up and spin-down components of the potential (which models in this case applying an external magnetic field).
  - *ispl*=1: with permanent splitting (on each iteration); then *split* is the splitting between spin-up and spin-down components of the potential in Ry. This is done according to:

$$V_\sigma(\mathbf{r}) + \frac{1}{2}\delta V(-1)^\sigma \tag{1}$$

where δV is the splitting and σ runs from 1 to 2 for spin-up and spin-down components respectively. Note then if *ispl*=0 but *split* is not zero, the latter will be used only at the first iteration. Specify *ispl*=0 and some non-zero splitting when doing spin-polarized calculation and starting from NON-spin-polarized charge density. Then, the input charge density file contains only charge density and no spin densities, the potential for spin-up and spin-down states is equivalent and must be splitted to push the system out of the paramagnetic solution. The magnetization density will be artificially introduced after the first iteration. At the following iterations *split* will be set to zero automatically and if the system tends to be magnetic, the self-consistent procedure should converge to it. If one continues self-consistency starting from the SPIN-POLARIZED charge density, then non-zero splitting in INIFILE will be ignored. This is done in order to perform smoth continuation of the self-consistent procedure from one run to another run. It might happens, however, that it is useful to make the splittting at the first iteration even if the input charge density is spin polarized. For example, the system is too far from the self-consistency or the previous calculation was done non-magnetic but spin-orbit coupled. (In the latter case the charge density file containes both spin-up and spin-down components which are equivalent). For this purpose specify *ispl=2*. This will suppress setting *split*=0 at the first iteration.

- mass: atomic mass as in the periodic table. This value is read but not used by the NMT programs.

- *mt-sphere*:
  In the ASA version, this value is ignored.
  In the CEL version, this is the value of the non-overlapping sphere which must lie completely inside the polyhedron.
  In the PLW version, this is the non-touching muffin-tin sphere radius.

- *rou-sphere*:
  In the ASA and PLW versions this value is ignored.
  In the CEL version, this is the radius of a circumscribed sphere around the atomic polyhedron, or slightly larger. The range between MT sphere and the polyhedron boundary represents the interstitial region of a given atom while the spherical harmonic expansions are performed for the charge density and the potential in the region between inscribed (MT) and circumscribed spheres. Depending on the shape of the polyhedron these spherical harmonic expansions can be slowly convergent. For cubic BCC or FCC lattices the charge density is well convergent with l_{max}=8. For the diamond structure, on the other hand, there are some points in the cell where one-center spherical harmonic expansion cannot be performed. In this case one has to introduce empty spheres to improve close packing.

- *weight*: initial weight factor for blowing up atomic spheres.
  In the ASA version, this is the weight with which the atomic sphere will be blown up so that the space-filling occurs. Only relative values of weights of different atoms play a role, as they are properly normalized inside the program according to the formula:

$$s_\tau = \left(\frac{\Omega_{cell}}{\sum_\tau \frac{4\pi}{3} w_\tau^3} \right)^{1/3} \times w_\tau \tag{2}$$

where s_τ is the atomic sphere radius, w_τ is the input weight, and Ω_{cell} is the cell volume, . The actual values of the atomic sphere radii are printed out in the output file.
In the CEL version, this parameter is used to find *mt-sphere* and *rou sphere* radii. Here weights are used for *blowing up the polyhedra.* Since one can imagine that there are fat and thin atoms then they can be described with big and small polyhedra. Again only relative fatness of every atom is important, the division of the space into different polyhedra will always conserve the total volume. After the polyhedra are found, the program will print out the radii of inscribed and circumscribed spheres in the output file and these radii must be put back into the INIFILE.

In the PLW version, the weights also help to find the *mt-sphere* radii. The spheres around every atom will be blowed up until they touch. The speed of blowing the spheres is proportional to the weight specified. This again allows to control fatness of every atom.

- rloc: localization radius. This large sphere around given site gives all nearest neighbors used for builiding screened LMTOs in TB -calculations. The parameter has no effect in all NMT programs, but can be used in some applications (see, e.g. chapter ADDITIONAL INPUT: HOPFILE).

Both CEL and PLW programs actually work only with the sphere radii while the weights are auxilary input data. The strategy for choosing the spheres is the following. First, one chooses the 'weights' of different sorts of atoms from some physical arguments (atomic sizes, lattice geometry). Then, one runs a special program located at */nmt*/lib/cell/* . The program reads INIFILE and generates atomic cells according to the weights specified. The output information will contain the muffin-tin spheres and circumscribed spheres found in this way which should be placed back into the INIFILE. See the description of */nmt*/lib/cell/* program in the chapter USING LOCAL LIBRARY for more details.

- *lmax-t*: maximal angular momentum l (not $l+1$!) for the basis functions (i.e. for the decomposition of the tails coming from other atoms). Normally it is 4 in the ASA calculation, 6 for PLW calculation and 8 for the CEL calculation.

- *lmax-b*: maximal l actually included in basis. This may not be less than number of non-zero columns in the description of any valence states (see below).

- *lmax-v*: maximal l-value for the expansion of the charge density and the potential in spherical harmonics. Normally it is the same as *lmax-t*.

The following block is repeated for each κ from 1 to *nkap*:

```
......
Valence states are:
 s p d f                        - states for E=-0.6 Ry
 4 4 3 4                        - main quantum numbers
 0 0 0 0                        - basis set
 0 0 0 0                        - choice of Eny
-0.5,-0.5,-0.5,-0.5             - Eny
-1.0,-2.0,-3.0,-4.0             - Dny
 s p d f                        - states for E=0.4 Ry
 4 4 3 4                        - main quantum numbers
```

```
 1 1 1 0                        - basis set
 3 3 1 0                        - choice of Eny
 0.5, 0.5, 0.5, 0.5             - Eny
-1.0,-2.0,-3.0,-4.0             - Dny
 s p d f                        - states for E=1.4 Ry
 4 4 3 4                        - main quantum numbers
 1 1 1 0                        - basis set
 3 3 1 0                        - choice of Eny
 0.5, 0.5, 0.5, 0.5             - Eny
-1.0,-2.0,-3.0,-4.0             - Dny
.......
```

- *s p d f - states for E=0.4 Ry* . This is a comment.

In each of the following lines, only (*lmax-b* + 1) first numbers are read (the only exception is the *Eny* line where (*lmax-b* + 2) numbers are read; see below). For instance, if *lmax-b*=2 and you define something for the f-states, this will have no effect.

- *4 4 3 4*: - main quantum numbers
- *1 1 1 0* - basis set; either 1 (if the state is included in basis), or 0. In the example for NiO, 2κ basis set is chosen for Ni s, p, and d-states with the tail energies 0.4 and 1.4 Ry. Usually the distance between energies is 1 Ry or so to avoid linear dependency of LMTOs of the same l-character but different by κ^2. The same 2κ basis is chosen for the broad 2p-states of oxygen. On the other hand narrow 2s oxygen states are treated with 1 κ^2=-0.6 Ry.
- *3 3 1 0* - choice of E_ν; for each state, it may be:
  - 0: E_ν is taken from the *Eny* line and fixed throughout the iterations
  - 1: D_ν is taken from the *Dny* line and fixed throughout the iterations. E_ν are adjusted to these *Dny*.
  - 2: E_ν is found as the energy of the bound state inside the atomic sphere. Advisable for the states which are semicore like but treated as bands in the main valence panel. If the bound state cannot be found (the eigenvalue is lying in the continious spectrum), the E_ν will be fixed according to $D_\nu = -l - 1$
  - 3: E_ν is adjusted throughout the iterations to the center of gravity of the occupied band from the band structure calculation; at the first iteration, however, the D_ν from the *Dny* line is used, and Eny ignored. During the self-consistency, the *Eny* numbers will be stored to the SCFFILE. In case of restart calculation, they will be read from it to obtain a smooth continuation.
  - *0.5, 0.5, 0.5, 0.5 - Eny*
  - *-1.0,-2.0,-3.0,-4.0 - Dny*

It should be noted that in the Eny line, (*lmax-b* + 2) numbers are read. For the states which are not included in basis but used in the decomposition of the tails coming from other atoms, the (*lmax-b* + 2)-th value in the *Eny* line is used as the energy of radial functions with higher l's in terms of which the decomposition is performed. The prescription of chosing this value is to place it into the center of

gravity of the whole occupied part of the band. (This parameter is not crutial for the calculation, a good estimate for it is the middle between bottom of the band and the Fermi energy.)

General prescription for choosing tail energies: for the programs working with the MT-geometry (CEL,PLW) the interstitial region can be large, therefore an additional variational freedom of the basis functions is desired. For the states forming broad energy bands 2 or 3 kappa basis set must be chosen to make sure that the result of the band structure calculation is well convergent. The values for these kappa's are not that important, the only condition is that they should be separated from each other by the energy of the order 1 Ry to avoid linear dependency of the LMTOs. A few hints can be given here: (i) negative–kappa basis set is formed with $\kappa_1{}^2$=-0.1 Ry, $\kappa_2{}^2$=-1.0 Ry, and $\kappa_3{}^2$=-2.5 Ry. The advantage of the negative energies is that they allow to avoid singularities of the structure consntants connected with the free–electron poles. (ii) positive–kappa basis set is formed with the first κ^2 placed in the center of gravity of the occupied band, another two kappa's are placed with the step 1 Ry above, i.e. $\kappa_1{}^2$=0.4 Ry, $\kappa_2{}^2$=1.4 Ry, and $\kappa_3{}^2$=2.4 Ry. Note that for the positive kappa case small imaginary part (0.03 Ry or so) must be added to avoid singularities in the structure constants.

Positive kappa basis set reminds an expansion over plane waves while negative kappa basis looks closer to the LCAO (linear combination of atomic orbitals) like representation. No specific preference to either chose can be given exept that the negative-kappa basis should be more stable numerically (no singularities in the structure constants).

For the narrow energy bands (semicore like) one can use single kappa basis set with the tail energy equal to the kinetic energy of that orbital in the interstitial region, i.e. $\kappa^2 = E - V_{int}$ where E is the position of the narrow band and V_{int} is the average potential in the interstitial region. When both these values are given with respect to the MT-zero (the potential at the MT-sphere averaged over different atoms) the value of V_{int} is usually small and a good estimate for the κ^2 is just the position of the band relative to the MT-zero. (That is actually done in the programs, all energies are given with respect to the MT-zero.)

For the ASA version usage of the multiple kappa basis is not important, single kappa basis is always OK. The tail energy can be fixed to 0 (or slightly smaller value, -0.1 Ry, to avoid structure consnant singularity) as was proposed in the original paper [1].

After the blocks distributing the valence states over different κ's, follows the description of the semicore panels:

```
......
semicore states are:
 1                              - # of semicore states
 3 1 (-4.0,0.0)                 - n,l, energy for 3p state
......
```

- 1: number of semicore states which will be treated as band states in separate energy windows (without hybridizing with the main valence panel)
- 3 1 (-4.0,0.0) : principal quantum # n, angular momentum l, tail energy κ^2 (here: 3p state) for every state.

This is self-explanatory, and it should be only noted that a possibility is provided that semicore states from different atoms belong to the same panel (and are therefore allowed to hybridize); for this, one should indicate the same tail energy for them. Semicore states with different tail energies are

treated as independent. During the iterations, E_ν's for the semicore states are chosen to be the bound energies for the spherical part of the potential for a given iteration.

Here the block *Atomic Data* is finished. The last block is

3.5 Output Control Parameters:

```
...
Output Control Parameters:
 2 'nio.con'                    - icon,<confile>
 0 'nio.srf'                    - isrf,<srffile>
 0 'nio.psi'                    - ipsi,<psifile>
 0 'nio.scr'                    - itmp,<tmpfile>
 0 'nio.bnd'                    - ibnd,<bndfile>
 0 'nio.pot'                    - ipot,<potfile>
 0 'nio.fat'                    - ifat,<fatfile>
 0 'nio.dos'                    - idos,<dosfile>
 2 'nio.scf'                    - iscf,<scffile>
 2 'nio.out'                    - iout,<outfile>
...
```

This section explicitly gives the names of files which may be used by the program. Generally, (but with a few exceptions) the switch in the first position has the following meaning:

- 0 - file is not created, or created as temporary.
- 1 - file will be opened as a new one and saved, if file exists, the execution will be terminated.
- 2 - file already exists and will be read, if file does not exist, the switch will be automatically set to 1.
- 6 - the contents of the file will be printed out to the terminal. (channel #6)
- *icon,<confile>*: file for storage the structure constants. Operative with all NMT* programs. If *icon=0*, CONFILE will be created as temporary in the scratch directory, the structure constants will be calculated for every k-point and at every iteration. This might save some disc space since no storage of the structure constants for all k-points is performed. If *icon=1* CONFILE will be created as permanent and the structure constants will be calculated and stored in this file for all k-points. If CONFILE exist, the program will terminate. Setting *icon=2* is more general option. If CONFILE does not exist, it will be created (the program will automatically set switch *icon=1*), if CONFILE already exist, the stored information will be checked for matching to the current set-up (such as # of k-points, tail energies, etc) and the structure constants will be read. Setting *icon=2* is advised in most cases.
- *isrf,<srffile>*: file for storage the cell surface constants which are necessary to integrate over the polyhedra. This file is only used by the CEL program. The meaning of the key *isrf* is the same as for the CONFILE except that there is no k-point dependence on the size of this file. Setting *isrf=2* is advised in most cases.

- *ipsi,<psifile>*: file for storage the wave functions. In fact only the coefficients in the expansion of the wave function into the LMTO basis are stored. The wave functions might be necessary for applications. The meaning of the key *ipsi* as above. Usually *ipsi=0*. No restart from this file can be performed.

- *ibnd,<bndfile>*: file for storage the energy bands. Setting *ibnd=1* allows to calculate the energy bands along high-symmetry directions (the latter ones are listed in the STRFILE, see below). To control how many k-points is necessary to generate along every direction, use first switch in setting the k-point sampling (see below in this chapter). The program will automatically set # of iterations equal to 1 and will stop after calculating the energy bands. Here some care should be taken with the CONFILE. Since the k-point sampling used for high-symmetry directions is different from that used for integrated over the BZ, the CONFILE must be either recalculated or you must specify *icon=0* when using the switch *ibnd=1*. The bands can be plotted using the program **plobands.exe** located in */nmtlib/plot/*. See chapter USING NMTLIB LIBRARY for details.
  Setting *ibnd=2* is another option which allows to store the energy bands in the tetrahedron mesh of the BZ. Since the bands are stored at the end of the self-consistent cycle, keep it in mind and set # of iterations to 1 manually if necessary. This storage can be necessary to plot the Fermi surface or to calculate real/imaginary parts of susceptibilites. See chapter USING NMTLIB LIBRARY for details.

- *ipot,<potfile>*: Specify *ipot=1 or 2* to store the potential into the POTFILE. The potential can be plotted afterwards using the program located at */nmt*/lib/grid/* . See chapter USING LOCAL LIBRARY for details.

- *ifat,<fatfile>*: Specify *ifat=1* together with the switch *ibnd=1* to store the information about the partial orbital character of the bands. This is extremily usefull to understand the chemistry of the compound. The fat bands can be plotted with help of the tight-bindindg LMTO-ASA programs developed in Stuttgart. FATFILE must be rewritten into the format understandable by the TB-LMTO using the program */nmtlib/plot/***fat.exe**. See chapter USING NMTLIB LIBRARY for details.

- *idos,<dosfile>*: Specify *idos=1* to calculate the density of states. Here a starting DOSFILE should first be prepared, containing one line with the lower and upper energy limits and the number of divisions, like, for instance,-0.6 0.8 400. Then, in the last iteration partial densities of states are put in the same file (for 400 energies, in the example above). As soon as there is no well-justified way to define partial l-decomposed densities of states in the full potential calculation, what is actually calculated is the total density of states per unit cell decomposed proportionally to (n, l) contributions within the muffin-tin spheres. The density-of-states can be plotted using **plotdos.exe** program located at */nmtlib/plot/* directory. See chapter USING NMTLIB LIBRARY for details.

- *iscf,<scffile>*: This is the file for storing the output charge density. The file is updated at the end of every iteration. Since there are now two SCFFILEs, one for the input charge density and another is for the output charge density, they must have different names in order to avoid rewritting of the input file. Usually, the input charge density file has an extension **.sc0** (**nio.sc0** in the example for NiO) while the output SCFFILE has an extenison **.scf** (**nio.scf** in the example

for NiO). Specifying *iscf*=0 will not produce the output SCFFILE (Actually, it will be produced and erased at the end of the execution). Specifying *iscf*=1 will produce the output SCFFILE. If the file with the same name already exists the program will terminate. Specifying *iscf*=2 will produce the output SCFFILE regardless if the file with the same name exists or not (if it exists, it will be replaced). A usefull hint here is always to put *iscf*=2, and set the output SCFFILE to **nio.scf**. The input SCFFILE is **nio.sc0**. If the execution is sucessful, just rename **nio.scf** to **nio.sc0**, i.e set new input SCFFILE to the output SCFFILE. If the execution fails by some reason (mistake in the input, bad choice of the basis set, ghost bands, etc) do not bother to erase bad **nio.scf** (which will be created after the run), correct input data files and start the job again specifying **nio.sc0** as the input SCFFILE. Switch *iscf*=2 will automatically erase bad **nio.scf** file.

- *iout,<outfile>*: This is the file for storing current output at the iteration, printing total energy, forces, atomic charges, magnetic moments and a lot of other usefull information. If *iout*=0 this file will not be created (actually created and deleted after the execution is completed). If *iout*=1, the OUTFILE will be created and the output information stored. In case the file with the same name already exists, the program will terminate to avoid unnecessary rewritting. If *iout*=2, the OUTFILE will be created regardless if the file with the same exists or not. (If the file exists, it will be replaced by the new OUTFILE). An example how it works is the following. Suppose one makes 10 iterations. The **nio.out** file is created. If another 10 iterations are required, another **nio.out** file must be created. If *iout*=2, the old OUTFILE will be replaced by the new one. If *iout*=1, the program will terminate and you have to remove old **nio.out** manually. A useful hint here is (i) see the OUTFILE after the execution, (ii) decide whether the execution was OK or some mistake occurs, (iii) if it was OK, rename **nio.scf** to **nio.sc0** and **nio.out nio.ou0**. if it was not OK, remove bad output **nio.scf** and **nio.out**, correct input data and start the job again. This always allows to keep the switches *iscf=2* and *iout=2*.
  The last option with the OUTFILE is *iout>2*. In this case the output will be redirected to the channell assigned to the terminal output. Specify *iout=6* for the terminal output under UNIX systems. Specify *iout=5* for the terminal output under VMS systems. The output from the program is made using standard WRITE(UNIT=IUN,...) where IUN=1 when *iout<3* and IUN=IOUT when *iout*> 2.

3.6 Other Data for NMT-pack

```
....
Other Data for NMT-Pack:
-1,4                          - nff,nef
 8,8,8,2                      - n1,n2,n3,nc
 32,32,32,0.02,0.04,1,5       - nfft1,nfft2,nfft3,epsR,epsG,kbz,rbz
....
```

- *nff* : number of filled bands in the main valence panel (above the semicore!). If nff=-1, this number will be determined automatically from the knowledge of the total valence charge and the guessed number of bands crossing the Fermi level (see below).

- *nef:* number of bands crossing the Fermi level or larger. This parameter is used for calculating the Fermi energy and DOS. If this number is not exactly known, or if empty bands should also

be taken into account (for plotting DOS above the Fermi energy), *nff* may be set smaller than the true number of filled bands. One can always use combination *nff*=0 and *nef*=total # of bands. This, however, may lead to unnecessary large storage of the expansion coefficients $A^{\mathbf{k}\lambda}_{L\kappa\tau}$ which are written on the scratch file for the bands crossing the Fermi level.

- *pole:* this value is read but not used by the NMT programs.

- *n1, n2, n3, nc*: divisions of the Brillouin zone along three directions for the tetrahedron integration (*nc* sets first division for semicore states, other two will be determined automatically). Every k-point is described by the set of three integers (k1,k2,k3) according to:

$$\mathbf{k} = \frac{k_1}{n_1}\mathbf{G_1} + \frac{k_2}{n_2}\mathbf{G_2} + \frac{k_3}{n_3}\mathbf{G_3} \tag{3}$$

  where $\mathbf{G_1}$,$\mathbf{G_2}$, and $\mathbf{G_3}$ are the reciprocal lattice vectors. For calculating the bands along high-symmetry directions (switch *ibnd=1* described above) the first value of n1 will control the division of every direction (the number of k-points generated in every line is n1+1).

- *m1, m2, m3:* divisions of the unit cell for the fast Fourier transform. Every r-point of the FFT-grid is described by (i1,i2,i3) according to

$$\mathbf{r} = \frac{i_1}{m_1}\mathbf{R_1} + \frac{i_2}{m_2}\mathbf{R_2} + \frac{i_3}{m_3}\mathbf{R_3} \tag{4}$$

  where $\mathbf{R_1}$,$\mathbf{R_2}$, and $\mathbf{R_3}$ are the primitive lattice itranslations.

  The rule of thumb is, 16 divisions between nearest neighbours guarantees the sufficient accuracy. Another estimate (useful for complex structures) is, total number of divisions $m1 \times m2 \times m3$ should be not less than 4000 $\times$ the number of atoms, then distribute the divisions over three lattice vectors so as to get rouphly equiddistant mesh.

- *epsR, epsG* : accuracy of matching the spherical Hankel functions in real and reciprocal space. Always specify epsR=0.02 and epsG=0.04.

- *keyt, bzr:* to accelerate Fourier transforms when calcualting interstitial potential matrix elements, set *keyt*=1. In this case the radius of the cutoff sphere in reciprocal space is set by parameter *bzr* times the radius of the sphere circumscribing the Brilloiun zone. Usually it is 4-6. The smaller *bzr* value the faster calculation the lower the accuracy.

Because of the permanent development of different parts of the programs there is a need to provide an additional information. If this is necessary additional input files must be listed at the end of INIFILE. (There will be no mistake if the following lines are absent in the INIFILE)

```
....
Additional input files:
 2 'nio.hub'                    - ihub,<hubfile>
 0 'nio.hop'                    - ihop,<hopfile>
 0 'nio.opt'                    - iopt,<optfile>
 0 'nio.enr'                    - ienr,<enrfile>
 0 'nio.pnt'                    - ipnt,<pntfile>
....
```

- ihub,<hubfile>: Specify *ihub=1* if it is necessary to perform LDA+U calculations. See chapter ADDITIONAL INPUT: HUBFILE for the details.
- ihop,<hopfile>: Specify *ihop=1* if it is necessary to withdraw hopping matrix elements. See chapter ADDITIONAL INPUT: HOPFILE for the details.
- iopt,<optfile>: Reserved for calculating the optical properties by the package OPT (not currently avaiable).
- ienr,<enrfile>: This file is not used by the NMT programs.
- ipnt,<pntfile>: This file is not used by the NMT programs.

4 STRUCTURE CONTROL FILE: STRFILE

STRFILE describes the crystal structure. An example is given below:

```
     ****** Structure Data for NiO ******
-------------------------------------------------------
Main Parameters :
 4,  1.0 1.0                    - natom,b/a,c/a
 1 1 1 1                        - ibas, ibz, icalc,istrain
 4 10                           - ndiv1 ndiv2
 300 , 1.0                      - nvec_max,alpha
Primitive Translations :
 1/2 , 1/2 , 1.0                ! Ax,Ay,Az
 1/2 , 1.0 , 1/2                ! Bx,By,Bz
 1.0 , 1/2 , 1/2                ! Cx,Cy,Cz
Basis in original  Unit-Cell :
 0.0 , 0.0 , 0.0                ! Ni1
 1.0 , 1.0 , 1.0                ! Ni2
 1/2 , 1/2 , 1/2                ! O
 3/2 , 3/2 , 3/2                ! O
Basis in distorted Unit-Cell :
 0.0 , 0.0 , 0.0                ! Ni1
 1.0 , 1.0 , 1.0                ! Ni2
 1/2 , 1/2 , 1/2                ! O
 3/2 , 3/2 , 3/2                ! O
Strain Matrix:
 1.0 , 0.0 , 0.0                ! S1(1..3)
 0.0 , 1.0 , 0.0                ! S2(1..3)
 0.0 , 0.0 , 1.0                ! S3(1..3)
Point Group Description File :
 'C'                            - cubic system
Primitive translations for BZ :
 0.0 , 0.0 , 0.0
 0.0 , 0.0 , 0.0
 0.0 , 0.0 , 0.0
High-symmetry direction settings (Example):
 4                              - # of directions
 g-X
 0.0,0.0,0.0,  1/2,0.0,0.0
 g-L
 0.0,0.0,0.0,  1/2,1/2,0.0
 Z-R
 0.0,0.0,1/2,  1/2,0.0,1/2
 R-A
 1/2,0.0,1/2,  1/2,1/2,1/2
Settings for "plobands" :
 X-g-L,Z-R-A.
```

The meaning of parameters is as follows:

4.1 Main Parameters

- *natom*: total number of atoms in the unit cell
- *b/a, c/a*: orthorombicity parameters
- *ibas*: switch for atomic setting;
  - 1: basis vectors are given in Cartesian system
  - 0: basis vectors are given in units of primitive translations
- *ibz:* switch for Brillouin zone construction;
  - 1: Brillouin zone translations are set up automatically (as reciprocal lattice translations)
  - 0: BZ-translations are read from the corresponding lines in the STRFILE. The idea is, it may be useful if the automatic Brillouin zone has too pathological shape for dividing into tetrahedra.
- *icalc:* switch for using the calculator to translate the expressions.
  - *icalc*=1 - the calculator is "on" for interpreting the expressions in the sections "Primitive Translations", "Basis...", "Strain Matrix", "Brillouin zone", "High-symmetry settings". Every line can contain any simple expressions /,*,+,-, brackets are allowed without restrictions, special functions like COS, SIN, TAN, EXP, LOG, SQRT, CBRT=X**(1/3) are allowed in the format of FORTRAN, but nesting of the special functions is not allowed. Special constant PI=3.1415... can be specified. Degrees like (...)*(...) or (...)$\hat{}$(...) are also allowed, but the expressions in brackets cannot contain special functions listed above. Expressions are separated by commas, and at the end any comment only after ! is allowed! No letter case sensitivity is assumed. Look also at some comments in in the */nmt*/run/***calc.f** - file. Some simple examples are:

```
Primitive Translations for the hexagonal lattice:
 1.0 , 0.0             , 0.0    ! Ax,Ay,Az    !!! <-- note for the comment"!"
 1/2 , sqrt(3)/2       , 0.0    ! Bx,By,Bz    !!! <-- which is only allowed
 0.0 , 0.0             , 1.0    ! Cx,Cy,Cz    !!! <-- when ICALC=1

Basis in the hexagonal lattice:
 0.0 , 0.0             , 0.0    !  Zn
 1/2 , 1/2/sqrt(3)     , 1/2    !  Zn

Rotation of the coordinate system:
 cos(pi/4)    , sin(pi/4)    , 0.0
-sin(pi/4)    , cos(pi/4)    , 0.0
 0.0          , 0.0          , 1.00

Tetragonal strain c/a=1.10, conserving the volume :
 (1.10)**(-1/3),  0,              0
  0,             (1.10)**(-1/3),  0
  0,              0,             (1.10)**(2/3)
```

  - *icalc*=0 - give up the calculator, use standard READ(1,*) - statement
- *istrain:* switch for taking care of the charge density plane-waves expansion in the presence of strain:

- *istrain*=1 - a sphere in reciprocal lattice which selects plane waves to expand the charge density will be distorted to some ellipsoid if the strain matrix (specified below) is not unit matrix. This is usefull for the distortions like strains changing b/a, c/a ratio, etc because if one always chooses a sphere to select plane waves for the Fourier transform, during the distortions the number of plane waves is changed. The latter can in principle lead to some errors in the energy difference for two lattice configurations.
- *istrain*=0 - always use a sphere to select plane waves for the Fourier transform.

- *ndiv1:* parameter connected with the polyhedron construction. This value is changed from 4 to 1. If *ndiv1*=4 the polyhedron construction is very fast because not too many nearest neigborous are considered as the potential sites producing the polyhedron planes. The expirience shows that, in most cases, it is sufficient to use *ndiv1*=2. If, however, the program prints incorrect value of defined cell volume, set *ndiv1*=1. (The best way indeed is to try to find different atomic weights which define the polyhedron geometry). This parameter has only meaning for the CEL version, not for ASA or plane-waves versions.
- *ndiv2:* cell-surface grid parameter for taking interstitial integrals. Increasing this parameter makes the calculation of the interstitial cell integrals more accurate but also more time consiming. The recommended value is 20. This parameter has only meaning for the CEL version, not for ASA or plane-waves versions.
- *nvec:* average number of lattice vectors used in the Ewald summation. The program will take the number of vectors in reciprocal space approximately twice of *nvec* and the number of vectors in direct space half of *nvec (*see also next paragraph*).* The program does not check the convergency of lattice sums but prints out the accuracy of the calculation (see below) which is the relative cotribution to the sum going from the largest vectors.
- α is usually equal to 1. Since the ratio between numbers of generated vectors in reciprocal and direct spaces is fixed by four (see previous paragraph), the possibility is provided to scale this ratio by a factor of α. For example if $\alpha = 1/4$, the number of generated vectors in both spaces will be the same. This scaling does not change the accuracy of the lattice sums but may accelerate the calculation of those. Parameter $\alpha = 1$ was chosen from the condition of fastest calculation for system with the number of atoms of order 10 at the work station IBM-RISC 6000.

4.2 Primitive Translations

First three Cartesian components of the first vector, then the second etc., as (in subr. READSTR):

```
do Ivec=1,3
READ(1,*)(RBASR(I,Ivec),I=1,3)
enddo
```

4.3 Basis in original Unit-Cell

The same as above. In fact, this data are not currently in use, but may be used for programmer's information.

4.4 Basis in distorted Unit-Cell

These lines are really used to set the basis. Note that the atomic positions may be provided either in Cartesian coordinates (for ibas=1), or in the units of primitive translations (for ibas=0). In the output file, "true" Cartesian coordinates are printed out, after all transformations.

4.5 Strain matrix

This matrix performs the linear transformation of the translation vectors and the basis positions according to: R(new)=S*R(old). For reciprocal lattice the rule is G(new)=G(old)*S(-1). This may be convenient for introducing strain or rotating the whole setup to other axes. This matrix should always be provided; in the default case it is simply the unit matrix. The matrix is read in row by row, as:

```
      do I=1,3
      READ(1,*)(STRAIN(I,J),J=1,3)
      enddo
```

In restoring the "true" translation vectors and basis positions, the program *first* applies the orthorombic scaling:

```
      DO 10 I=2,3                            !   I=2 for b/a,  I=3 for c/a
      do IVEC=1,3
        RBASR(I,IVEC)=RBASR(I,IVEC)*ORTH(I)   ! direct lattice
        BBASR(I,IVEC)=BBASR(I,IVEC)/ORTH(I)   ! recipr. lattice
      enddo
      do IATOM=1,NATOM
        TAU R(I,IATOM)=TAU R(I,IATOM)*ORTH(I) ! basis positions
      enddo
  10  CONTINUE
```

and *then* the strain matrix:

```
      DO IVEC=1,3
        do I=1,3
        RBAS(I,IVEC)=0.D0
        do J=1,3
        RBAS(I,IVEC)=RBAS(I,IVEC)+STRAIN(I,J)*RBASR(J,IVEC)
        enddo
        enddo
      ENDDO
C  APPLY STRAIN FOR BASIS VECTORS:  T'=S*T
      DO IATOM=1,NATOM
        do I=1,3
        TAU(I,IATOM)=0.D0
        do J=1,3
        TAU(I,IATOM)=TAU(I,IATOM)+STRAIN(I,J)*TAUR(J,IATOM)
        enddo
        enddo
      ENDDO
```

and correspondingly for the reciprocal lattice.

4.6 Point Group Description

Symmetry code: either 'C' for cubic systems or 'H' for hexagonal systems. Cubic system contains 48 operations of a cube, hexagonal system contains 24 operations of a hexagon (it incudes rotations about 60 degrees along z-axe). Applying all symmetry operations for cubic or hexagonal group, the program picks up those which are consistent with the actual crystal structure. Non-symmorfic operations are found as well. Therefore your choice is only to decide whether a particular structure belongs to the cubic or hexagonal symmetry. Note that since C or H rotational operations assume a certain coordinate system (as rotation about 60 degress along z-axe, not

x or y axes) the same coordinate system should be used to describe crystalline structure. (For example, HCP structure cannot be described in the coordinate system with the rotations about 60 degrees along x-axe.) If it is necessary to use another rotational system, set symmetry code to 'A' (arbitrary) and give the file name of KOVFILE describing your own rotational system afterwards. See chapter ADDITIONAL INPUT:KOVFILE for details.

4.7 Primitive Translations for BZ

For *ibz*=1, these vectors are ignored. If *ibz*=0, the whole Brillouin zone is represented as a parallelepiped, the three primitive translations of which are set in this lines. The parallelepiped is divided along three directions (number of divisions is specified in the INIFILE then the group symmetry operations are applied to sort out non-equivalent **k** points. The Brillouin zone should correspond to the translation vectors as defined in "Primitive Translations" *prior* to applying eventual orthorombic displacement and strain to the latter. In this case, the program will automatically introduce the correct distortion of the Brillouin zone.

4.8 High-symmetry direction settings

are used for calculating the energy in a set of directions listed here. The program will do it if the parameter *ibnd* is set to 1 in the list of output parameters of the INIFILE. Note that if it is necessary to output the energy bands at the tetrahedron corners, set parameter *ibnd*=2.

4.9 Settings for PLOBANDS program

Not used by the fp-lmto code. This is a list of symbols which defines how to plot the energy bands using */nmtlib/plot/***plobands.exe** utility. See chapter USING NMTLIB LIBRARY for details.

5 INPUT CHARGE DENSITY FILE: SCFFILE

The charge density distribution in the unit cell is stored in the unformatted SCFFILE. This is the third input file to the NMT package. Since the representation of the charge density is different for the ASA, CEL or PLW programs, this file is unique for each of it. That means that you cannot start execution, say, of the PLW package giving, as an input file, the output SCFFILE made by the ASA program.

To prepare the input SCFFILE at the begining, there exists an auxilary program SCFM located inside */nmt*/lib/scfm/* directory. It uses Mattheiss's prescription for constructing the input charge density as a superposition of the atomic charge densities. To be able to do it, the directory */atomdat/* must exist and contain a library of the atomic charge densities for all elements of the periodic table. The */atomdat/* directory is distributed together with the NMT-package.

To properly configure SCFM, (i) go to */nmt*/lib/scfm/* directory, (ii) edit the file **makescfm.f** (iii) locate a string

```
C HERE IS THE PATH TO THE ATOMIC-DATA DIRECTORY:::
        DIR='/u/a4/savrasov/atomdat/'
```

(iv) change this path to your own path, (v) compile, link all the files and get an executable **make.exe** file.

To use the , you also need INIFILE and STRFILE to be prepared, because for the proper execution the crystalline structure, lattice parameter, the atomic charges, and the MT-sphere radii of the atoms must be known.

To run the program, goto the directory in which the input data are stored (like */nmt*/dat/nio/*). Then, issue the command: */nmt*/lib/scfm/***make.exe**. An input to the program is INIFILE, STRFILE, and the localization radius in atomic units. The latter is the radius of a sphere within which the summation of tails of atomic densities to a given site will be performed. Usually it is 10 a.u. Also you will be asked by the SCFFILE which is the output from the program. Specify the name like, e.g., **nio.sc0**. The SCFFILE will be stored in the current directory.

6 RUNNING THE PROGRAM

The full-potential program asks for the names of three files:

INIFILE?
SCFFILE?
STRFILE?

One has either to answer this from console, or to put the answers in a file, for example:

>**nohup** */nmtplw/run/***main.exe** < **nio.job&**

where **nio.job** is:

nio.ini
nio.sc0
nio.str

or to redirect the input stream:

>**nohup** */nmtplw/run/***main.exe** <<'EOF' &
>**nio.ini**
>**nio.sc0**
>**nio.str**
>**EOF**

After the program finishes, the output file **nio.scf** may be renamed to **nio.sc0** (OUTFILE may also be renamed), and the program can be run again.

After shifting the atoms, changing the k-point sampling, changing or using another tail energies κ^2, using another number of semicore panels, CONFILE should be recalculated if it was stored before. If it is necessary to change the sphere radii, the calculation may be continued from the previous SCFFILE. All needed interpolations to another meshes are done automatically inside the program.

Specifically for the CEL package: If one changes the sphere radii, the surface structure constants (SRFFILE) should be recalculated if they were stored before.

7 OUTPUT MESSAGE FILE: OUTFILE

Here, description of the output messages is given. Also short introduction to the structure of the program is described (see Figure 2).

Consider one particular iteration for NiO system by the program NMTPLW. The structure and the output messages of NMTASA and NMTCEL are similar.

7.1 Reading input data

The execution of the NMTPLW package (source file **main.f**) starts from reading the input data controled by INIT subroutine (see file **init.f**). Beginning of the OUTFILE contains the information read from the INIFILE.

```
<<<--- INPUT INIFILE READ --->>>

     *** Band Structure Calculation of NiO  ***
-----------------------------------------------------
<CONTROL PARAMETERS>    :
lift  = 3               - Self-consistency is on
lmto  = 1               - Unscreened  LMTO is on
novr  = 2               - Overlap matrix via Sdot
npf   = 0               - Atomic forces   are off
<EXCHANGE-CORRELATION>  :
ixc   =+3               - Janak-Williams-Morruzi, no GGA
<CONVERGENCY DATA>      :
niter = 50              - # of iterations
epstot=  .1000000E-08   - total energy   convergency
epsrho=  .1000000E-09   - charge density convergency
epsmag=  .1000000E-09   - magnetization  convergency
lbroy =  2              - Broyden mixing is switched
ibroy =  1              - after  n=ibroy  iterations
admixb=  .3000000       - initial guess for Jacobian
admixh=  .3000000       - mixing for high-l (>lbroy)
<ATOMIC DATA>           :
natom =  4              - # of atoms in unit-cell
nsort =  3              - # of atoms of different type
par   =  8.051000       - lattice parameter in a.u.
nspin = 2               - including spin-polarization
norbs = 1               - without spin-orbit coupling
nkap  = 3               - # of kappas in valence panel
Etail1= -.60000         - Hankel tail energy
Etail2=  .40000         - Hankel tail energy
Etail3=  1.4000         - Hankel tail energy
<OUTPUT DATA>           :
icon  = 1               - save str. const. in    nio.con
iftr  = 0               - no storage fourier con.
ibnd  = 0               - no storage of bands
idos  = 0               - no d.o.s.calculation
ipot  = 0               - no storage of full potential
iscf  = 1               - save full density   in nio.scf
iout  > 1               - print current output
<OTHER DATA>            :
nff   = 14              - # of bands below    EF
```

```
 nef   =  4              - # of bands crossing EF
 ndiv  = 8  8  8         - tetr.mesh for valence  panel
 ndic  = 2  2  2         - tetr.mesh for semicore panels
 <ADDITIONAL INPUT FILES>:
 ichub  > 0              - with Hubbard correction after nio.hub
 ichop  = 0              - with no hopping matrix
 icopt  = 0              - with no optical properties
```

This information is printed only to test the corectness of the input data. The next output lines contain the information read from the STRFILE.

```
<<<--- INPUT STRFILE READ --->>>

      ****** Structure Data for NiO ******
 -------------------------------------------------
 <CONTROL PARAMETERS>     :
 natom =  4                - # of atoms in unit-cell
 b/a   =  1.0000           - orthorombicity  along b
 c/a   =  1.0000           - orthorombicity  along c
 ibas  =  1                - basis in Cartesian sys.
 ibz   =  1                - automatic BZ - choice
 icalc =  1                - using built-in calc.
 istrn =  1                - distort cutoff sphere
 ndiv1 =  4                - polyhedron   generation
 ndiv2 = 10                - polyhedron surface grid
 nvec  =  300              - vectors in Evald method
 alpha =  1.0000           - splitting factor  there
 <PRIMITIVE TRANSLATIONS> :
 (  .50000    ,  .50000    ,  1.0000    ) ! R1x,R1y,R1z
 (  .50000    ,  1.0000    ,  .50000    ) ! R2x,R2y,R2z
 (  1.0000    ,  .50000    ,  .50000    ) ! R3x,R3y,R3z
 <BASIS ATOMS IN CELL>    :
 (  .00000E+00,  .00000E+00,  .00000E+00) ! for Ni1
 (  1.0000    ,  1.0000    ,  1.0000    ) ! for Ni2
 (  .50000    ,  .50000    ,  .50000    ) ! for O
 (  1.5000    ,  1.5000    ,  1.5000    ) ! for O
 <STRAIN MATRIX>          :
 (  1.0000    ,  .00000E+00,  .00000E+00) ! Sxx,Sxy,Sxz
 (  .00000E+00,  1.0000    ,  .00000E+00) ! Syx,Syy,Syz
 (  .00000E+00,  .00000E+00,  1.0000    ) ! Szx,Szy,Szz
 <INVERSE STRAIN MATRIX>  :
 (  1.0000    ,  .00000E+00,  .00000E+00) ! Rxx,Rxy,Rxz
 (  .00000E+00,  1.0000    ,  .00000E+00) ! Ryx,Ryy,Ryz
 (  .00000E+00,  .00000E+00,  1.0000    ) ! Rzx,Rzy,Rzz
 <POINT GROUP DESCRIPTION>:
 ikov  = C  - Cubic system
 <RECIPROCAL LATTICE>     :
 ( -.50000    , -.50000    ,  1.5000    ) ! G1x,G1y,G1z
 ( -.50000    ,  1.5000    , -.50000    ) ! G2x,G2y,G2z
 (  1.5000    , -.50000    , -.50000    ) ! G3x,G3y,G3z
 <BRILLOUIN ZONE>         :
```

```
( -.50000    , -.50000    ,  1.5000    ) ! K1x,K1y,K1z
( -.50000    ,  1.5000    , -.50000    ) ! K2x,K2y,K2z
(  1.5000    , -.50000    , -.50000    ) ! K3x,K3y,K3z
Cell Volume =  260.9273
```

The primitive translations, positions of atoms in the basis, reciprocal lattice vectors and the Brillouin zone data are given after taking into account b/a and c/a ratios, and after applying the strain matrix. The data in direct space are given in the units of lattice parameter, the data in reciprocal space are given in units $\frac{2\pi}{a}$. The cell volume Ω_{cell} is printed in atomic units [3].

The following output contains the information about the number of plane waves to be used for the representaion of the charge density and the potential in the interstitial region. This value together with the energy cutoff is found according to the input FFT grid. (32*32*32 in this case).

```
Number of plane waves to be used    3862
Plane-wave energy cutoff will be   91.81580     Ry
Reciprocal lattice sphere radius   12.27803     2pi/a
Fast fourier transform divisions  32  32  32
FFT optimal divisions calculated  32  32  32
```

Note that the line "FFT optimal divisions calculated" is for your information only, if optimally calculated divisions do not coincide with the input divisions, place optimal divisions into the INIFILE.

The following output messages contain information about using the acceleration in calculating the interstitial potential matrix elements (see also a description of parameters *kbz,rbz* in the INIFILE). The last message here compares plane wave series read from the input SCFFILE and generated in this set-up. In the example, both plane-wave set-ups are the same.

```
# of (k+G)-terms in Fourier sums     154
Brillouin zone radius calculated   .8660254     2pi/a
Teilor"s sphere radius estimated   4.330127     2pi/a
Teilor"s cutoff energy estimated   11.41988     Ry
Plane waves: old=   3862; new=   3862; coinciding    3862
```

7.2 Preparing structure constants

After executing the INIT subrouitine, the execution transfers to the package of programs controlled by STRMSH (see source file **strmsh.f**). Here the data depending on the crystalline structure are prepared. Next lines give an information about number of the point group elements found for the original and distorted lattices (according to the positions of atoms for original/distorted lattices listed in the STRFILE). The number of k-points generated for the main valence panel and semicore panels is also printed.

```
******* STRMSH started  ; CPU time:   93.15000     ************
   12 elements discovered for original   lattice
   12 elements found      for distorted  lattice
   65 k-points generated  for main valence panel
    4 k-points generated  for semicore    panels
```

The following messages prints non-zero expansion coefficients of the spherical harmonics for the charge density which are allowed by symmetry.

```
Position : (  .00000E+00,  .00000E+00,  .00000E+00) for Ni1
LMTO-basis set is expanded in spherical harmonics up to Lmax= 6
Charge density is expanded in spherical harmonics up to Lmax= 6
```

```
 Non-zero elements allowed by symmetry are the following:
 l= 0 ; m=                            0
 l= 2 ; m=                      -2  -1   1   2
 l= 4 ; m=           -4  -3  -2  -1   0   1   2   3   4
 l= 6 ; m=   -6  -5  -4  -3  -2  -1   0   1   2   3   4   5   6
 Total # of non-zero components found  27

...
```

This message gives the accuracy in expanding (pseudo)Hankel functions (for different l's) in plane wave in the interstitial region for the input cutoff (estimated from the input FFT-grid). The ratio RH(S)/H(S) is the ratio between pseudoHankel function and the real Hankel function at the sphere boundary. The pseudoHankel function should coincide with the Hankel function in the interstitial region but it is smooth inside the sphere (regular at the origin). The ratio GH(S)/RH(S) is the ratio between the pseudoHankel function found using Fourier transform and calculated in the real space. Diviations show the accuracy. The accuracy is worsen for higher l's but is acceptable (usually a few percent).

```
  l=    Nplw   Ecut (Ry)   RH(S)/H(S)    GH(S)/RH(S)
 " 0"   3862    91.8       1.00000        1.0000
 " 1"   3862    91.8        .99997        1.0000
 " 2"   3862    91.8        .99978        .99969
 " 3"   3862    91.8        .99895        .99853
 " 4"   3862    91.8        .99642        1.0027
 " 5"   3862    91.8        .99036        1.0162
 " 6"   3862    91.8        .97841        1.0063

...
```

The information below contains the set-up for using the Ewald method to sum up the structure constants. Minim differences between tail energies and the poles of free-electron Green function show how far is the singularity on the real axe. Note that when using the positive κ^2 a small imaginary part (approximately 0.03 Ry) must be placed to avoid this singularity.

```
 Result from VECGEN for direct/reciprocal spaces --->
 Rmax=  2.616219    ; Accuracy=  .2947387E-12; # of vectors=   163
 Gmax=  6.592308    ; Accuracy=  .4712605E-17; # of vectors=   609
 Smax=  4.473601    ; Accuracy=  .1545859E-19; # of vectors=   759
 Min.energy for using Evald"s method= -2.262885     Ry
 Total # of connecting vectors found       7
 Minimum difference between |k+G|^2 and kappa1^2 is   .9851246
 Minimum difference between |k+G|^2 and kappa2^2 is   .1121901E-01
 Minimum difference between |k+G|^2 and kappa3^2 is   .1749081E-02
```

7.3 Computing sphere radii

After preparing the structure constants, the execution of the program transfers to the package of programs for finding atomic polyhedra and different sphere radii. This task is controlled by the program CELLS (see source file **cells.f**). Next lines in the OUTFILE contain the information about accuracy in using the Fourier transforms for the pseudoLMTOs in the interstitial region. The pseudoLMTOs are defined as linear combinations of the

Hankel functions in the interstitials but they are smooth inside the spheres. (In other words pseudoLMTOs are linear combinations of pseudoHankel functions). The number of plane waves using for Fourier transform of every "s", "p", or "d" partial wave is different (the higher l the more cutoff is required). The ratios RH(S)/H(S) and GH(S)/RH(S) show the accuracy (see explanation above). This accuracy is regulated by the input parameters *epsR* and *epsG* in the INIFILE.

```
******* STRMSH finished ; CPU time:   94.08000     ************
******* SCELLS started  ; CPU time:   94.08000     ************
 Position : (  .00000E+00,  .00000E+00,  .00000E+00) for Ni1
 Basis  Nplw   Ecut (Ry)   RH(S)/H(S)    GH(S)/RH(S); Ekap=  .400    ; p#1
  "s"    336    17.5        .99665        .99933
  "p"    548    24.4        .99754        .99875
  "d"    934    35.8        .99522        .99316
 Basis  Nplw   Ecut (Ry)   RH(S)/H(S)    GH(S)/RH(S); Ekap=  1.40    ; p#1
  "s"    336    17.5        1.0013        1.0003
  "p"    548    24.4        .99405        .99682
  "d"    934    35.8        .99682        .99527
```

Different spheres found by the program are printed below. Here input mt-sphere means the MT-sphere read from the INIFILE, *this radius is actually used by the program.* The following lines for information only. Input as-sphere means the radius of the atomic sphere calculated according to the weight read from the INIFILE. Cell mt-sphere is the radius of the sphere inscribed into the polyhedron. The polyhedra are generated according to the weights read from the INIFILE. Cell as-sphere is the atomic sphere estimated according to the volume of the polyhedron. Touch.mt-sphere is the maximal radius of the spheres when they do not overlap. Circums.sphere is the radius of the sphere circumscribed the polyhedron. Other information gives *nrad=318* - number of radial points for the input MT-sphere, *znuc= 28* - nuclei charge, *zcor= 18* - deep core charge, *npln= 18* - number of planes found for this polyhedron.

```
 input mt-sphere is   2.100000     ; input as-sphere is   2.610231
 cell  mt-sphere is   2.108595     ; cell  as-sphere is   2.605776
 touch.mt-sphere is   2.108595     ; nrad=318,znuc= 28,zcor= 12
 circums. sphere is   3.486185     ; npln= 18
******* SCELLS finished ; CPU time:   94.37000     ************
```

7.4 Finding full potential

After printing out different sphere radii, the self-consistent cycle starts. The self-consistency is controlled by the program SCF1 (see source file **scf1.f**). At the beginning of each iteration, first, the full poteitnal is calculated. As a result, the table below is produced in the OUTFILE. It should be noted that the boundary values of potential, $V(S)$, are given with respect to the "vacuum zero", *i.e.*,to the energy zero of the atomic program. Once the average $V(S)$ is found, energy zero is put there, and since that the items "Average potential in the interstitial region", Kappa's and band energies are given with respect to it. It is recommended to adjust the MT-radii in such a way as to make, if it is possible, the boundary potential values $V(S)$ above not very different for different atoms. The V and P values stand for the potential and pseudopotential while RO and PD values denote density and pseudodensity. M(S) is the magnetic moment within the MT-sphere, PM(S) is the pseudomagnetic moment (has no physical meaning. Notation S is for the sphere while 0 is for the atom origin.

```
                ITERATION   1

*******  VFULL started  ; CPU time:   97.46000     ************
 Input data for Ni1    in the position   1 ------->
 V-up(S)= -.4008890      | RO-up(S)=  .1974167E-01
```

```
 P-up(S)= -.4039995      | PD-up(S)=  .2012848E-01
 P-up(0)= -11.49737      | PD-up(0)= -.1143829E-04
 V-dn(S)= -.3899292      | RO-dn(S)=  .1860992E-01
 P-dn(S)= -.3910340      | PD-dn(S)=  .1878650E-01
 P-dn(0)= -11.50916      | PD-dn(0)=  .2230109E-03
    M(S)=  1.787173      |    PM(S)=  .4962198
...

 Average potential over the sphere boundaries is  -.3368602
 Average potential in the interstitial region is  -.9289500E-01
 Total charge in the interstitial region must be   3.494624
 Total charge found  via  fourier  transform  is   3.494624
 Auxilary density renormalization coefficient is   .9999460
 Magnetization  in  the  interstitial region  is  -.8180455E-09
 Total magnetization found in elementary cell is   .2444900E-10
```

7.5 Calculating energy bands

After constructing the full potential the execution of the program goes to the package of program for solving the eigenvalue problem of the LMTO method. It is controlled ba the program BANDS (see source file **bands.f**). Information about choice of Eny is prited below. Eny, Dny stand for the E_ν, D_ν values used for this particular l-channel, Cny is the estimated center of the l-band, Wny is its width. All the estimates are done using potential parameter relations.

```
*******  VFULL finished ; CPU time:   137.8900     ************
******** BANDS started  ; CPU time:   137.8900     ************
 3kappa spin-up panel # 1 : Band Structure Calculation of E(k) with :
     Eny:         Dny:         Cny:         Wny:     Et=  .400     for Ni1
   .40738      -.30297       .90466       6.4873     for 4s-state, center
   .48718       .62592       2.4811       2.7362     for 4p-state, center
   .60077      -3.0000       .60077       .39204     for 3d-state, fix.D
.....
 1kappa spin-up panel # 2 : Band Structure Calculation of E(k) with :
     Eny:         Dny:         Cny:         Wny:     Et= -4.00     for Ni1
  -3.9571      -5.3387      -3.9856       .70398E-01 for 3p-state, bind.E
     Eny:         Dny:         Cny:         Wny:     Et= -4.00     for Ni2
  -3.7966      -5.2605      -3.8260       .74285E-01 for 3p-state, bind.E
......
```

7.6 Integrating over Brillouin zone

After the band structure calculation finished, the Brillouin zone integration starts (controlled by BZINT, see source file **bzint.f**). The information below contains the Fermi energy found (EF), number of states calculated (TOS) which must coincide with the total valence charge. DOS stands for the density of states. Numbers of fully filled bands and the number of bands crossing the Fermi level is for information only. They will not overwrite the input numbers *nff, nef* from the INIFILE. Calculated plasma frequencies (zero in semiconductors) are also printed. At the end energy bands at the Γ-point are given.

```
******** BANDS finished ; CPU time:   305.5700     ************
******** BZINT started  ; CPU time:   305.5700     ************
 Semiconducting band structure discovered --------->
 dielectric gap value is   .2190656     Ry ;   2.980563     eV
```

```
 EF,TOS,DOS==   .7062796       32.00000       .0000000E+00
 EF,TOS,DOS==   .7062796       32.00000       .0000000E+00
 Calculated average square of electron velocities:
 <Vx^2>=  .0000000E+00 ; <Vy^2>=  .0000000E+00 ; <Vz^2>=  .0000000E+00
 Calculated bare plasma frequencies (in eV):
 om_p^x=  .0000000E+00 ; om_p^y=  .0000000E+00 ; om_p^z=  .0000000E+00
 # of fully filled bands =  16; input # =  14
 # of bands crossing Ef  =   0; input # =   4
 Energy bands at the Gamma-point for spin-up states are
  -.58822      -.48708       .21930       .23623       .23623
  .40584       .40584       .42354       .47532       .47532
  .55798       .65522       .65522       .65546       .65826
```

7.7 Constructing the charge density

When the charge density is calculated (program RHOFUL, see source file **rhoful.f**), the following output lines allow to check for the correct normalization. If it is more the a few per cent (more in the ASA), something is going wrong. If overlap matrix is not positive define or the ghos bands occur, the renormalization coefficient can strongly deviate from unity. Watch out then for the mistakes in the INIFILE.

```
******** BZINT finished ; CPU time:   354.7200     ************
******* ROFULL started  ; CPU time:   354.7200     ************
 Valence charge in whole  elementary cell  must be   44.00000
 Valence charge  found  via fourier  transform  is   44.10053
 Renormalization coefficient of the val.density is   .9977204
```

7.8 Renormalizing core levels

The renormalization of the deep core levels (program RENCOR, see source file **rencor.f**) for each atom results in the following output table:

```
******* ROFULL finished ; CPU time:   371.0600     ************
******* RENCOR started  ; CPU time:   371.0600     ************
 Orbital   n  l  j    # el.  Levels(Ry) for Ni1   ; Zcor=  12.000
  1s1/2    1  0  1/2    2    -600.7874
  2s1/2    2  0  1/2    2    -70.59245
  2p1/2    2  1  1/2    2    -61.48175
  2p3/2    2  1  3/2    4    -60.19977
  3s1/2    3  0  1/2    2    -6.755245
```

7.9 Evaluating total energy

The work of the program evaluating the total energy (program ENERGY, see source file **energy.f**) starts by printing the partial numbers and densities of states for every atom and for every energy panel. Different contributions to the total energy are printed afterwards.

```
******* RENCOR finished ; CPU time:   371.2300     ************
******* ENERGY started  ; CPU time:   371.2800     ************
 Occupation numbers for the  1th panel; <E>=  .3373072    ---->
 Summed over tail energies partial states for Ni1
 spdf... TOS =   .2261       .2236       7.846       .2214E-01
 spdf... MAG =   .2004E-02  -.2576E-03   1.762      -.3051E-04
```

```
 spdf up-TOS =   .1140       .1117       4.804       .1105E-01
 spdf dn-TOS =   .1120       .1119       3.042       .1109E-01
 spdf up-DOS =   .0000E+00   .0000E+00   .0000E+00   .0000E+00
 spdf dn-DOS =   .0000E+00   .0000E+00   .0000E+00   .0000E+00
.....

 ************************************************
 *       BAND ENERGY     =-35.824906474871      *
 *       CORE ENERGY     =-3584.3342723301      *
 *       POTENTIAL ENERGY=-10079.778848994      *
 *       KINETIC ENERGY  = 6459.6196701892      *
 *       COULOMB ENERGY  =-12559.261981630      *
 *       EXCH-CORR ENERGY=-276.39922038385      *
 *       HUBBARD ENERGY  =-115.65537813379      *
 *       TOTAL ENERGY    =-6491.6969099585      *
 ************************************************
```

7.10 Evaluating forces

The evaluation of forces (program FORCES, see source file **forces.f**) is the next step at the iteration. Note that the Hellmann-Feynmann forces are not accurate and large incomplete basis set corrections must be taken into account. (See the description of the input parameter *npfr* in the INIFILE).

```
 * CALCULATED FORCES AT THE CENTERS OF ATOMS ------>
 * Position --> (  .00000E+00,  .00000E+00,  .00000E+00) for Ni1
 * HF    FORCE: Fx=  .0000000E+00; Fy=  .0000000E+00; Fz=  .0000000E+00
 * CR    FORCE: Fx=  .0000000E+00; Fy=  .0000000E+00; Fz=  .0000000E+00
 * HF+CR FORCE: Fx=  .0000000E+00; Fy=  .0000000E+00; Fz=  .0000000E+00
```

7.11 Mixing charge densities

The last step at the iteration is mixing of the input charge density and the output charge density to prepare the input density to the next interation. The self-consistency of the charge density can be simply watched by comparing the input/output charges of the spheres at the iteration (Values S(inp) and S(out)). The same is done for magnetization, M(inp) and M(out). Values I(inp) and I(out) stand for the interstial charges. After Broyden mixing procedure the ne charge density and magnetization are constructed, the corresponding charges and magnetic moments within spheres are also printed out. A usefull parameter for watching whether the Broyden mixing is properly working is the iteration weight. At the begining this number is set to one. If the charge density with every iteration approaches the self consistency, the iteration weight is bigger than 1 and increased (can be up to infinity). If the iteration gives a bad guess for the charge density, the iteration weight becomes smaller than 1. If a few (say 10) consequent iterations gives the weight much small than 1 (of the order 0.1), than it is advised to switch off the Broyden mixing and use linear mixing instead.

```
******* ENERGY finished ; CPU time:   419.8300     ************
 Input/output charge transfer at the iteration  --->
     S(inp)              S(out)
  -1.701708           -1.701708      for Ni1
  -1.701708           -1.701708      for Ni2
  -.4833732E-01       -.4833723E-01  for O
  -.4833732E-01       -.4833723E-01  for O
 Input/output magnetic moment at the iteration  --->
     M(inp)              M(out)
```

```
  1.753985            1.753985      for Ni1
 -1.753985           -1.753985      for Ni2
 -.1958433E-11        .5435652E-11  for O
 -.1955769E-11        .5435652E-11  for O
    I(inp)              I(out)
  3.500091            3.500091
  3.500091            3.500091
***************************************************************
 Broyden Mixing for Rho(r), Iter-Weight=  1.091346
    Charge:        Magnetization:
 -1.701708           1.753985      for Ni1
 -1.701708          -1.753985      for Ni2
 -.4833727E-01      -.2704503E-12  for O
 -.4833727E-01      -.2717826E-12  for O
 Interst. Charge after Broyden  = -3.500091
 Magnetization over MT-spheres  = -.1523093E-10
```

The last lines show the evaluation of the mean-square difference between input and output charge densities (magnetizations) at the iteration. An example below shows that the self-consistency is well reached. Usually convergency of the total energy up to 0.1 mRy corresponds to the charge density self-consistency of the order 10^{-2}.

```
 Charge Density Self-Consistency=  .1047040E-05    .1495354E-06
 Magnetization  Self-Consistency=  .9249301E-06    .1891719E-09
```

8 PARAMETER FILE: PARAM.DAT

The dimensions inside the program are defined using PARAMETER statements. All parameter statments are collected in one file called PARAM.DAT and are included in every program using INCLUDE statement. An example of PARAM.DAT is given below:

```
C ************ MAIN PARAMETERS OF THE ARRAYS *****************
        PARAMETER (NSPINMAX=2  ) ! SPIN-POLARIZED VERSION WITH SPIN-ORBIT COUP.
        PARAMETER (NPAT=7      ) ! MAX.NUMBER OF ATOMS IN UNIT-CELL
        PARAMETER (NSORTMAX=5  ) ! MAX.NUMBER OF DIFFERENT TYPE ATOMS
        PARAMETER (NSYMMAX=50  ) ! MAX.NUMBER OF NON-ZERO COMPONENTS IN RHO,V
        PARAMETER (LMAXV=6     ) ! LMAX FOR THE POTENTIAL AND CHARGE DENSITY
        PARAMETER (LMAXB=3     ) ! LMAX FOR BASIS SET
        PARAMETER (LMAXT=6     ) ! LMAX FOR TRIAL- AND WAVE-FUNCTION EXPANSION
        PARAMETER (NKAPMAX=4   ) ! MAX.NUMBER OF TAILS IN TRIAL FUNCTION
        PARAMETER (NDIMMAX=150 ) ! MAX.DIMENSION OF HAMILTONIAN
        PARAMETER (NPANMAX=5   ) ! MAX.NUMBER OF PANELS
        PARAMETER (NRADMAX=450 ) ! MAX.NUMBER OF POINTS IN RADIAL MESH
        PARAMETER (KMAX=400    ) ! MAX.NUMBER OF K-POINTS
C ************ PLANE WAVE DATA PARAMETERS ****************
        PARAMETER (NPLWMAX=40000)! MAX.NUMBER OF PLANE WAVES IN FOURIER TRANSF.
        PARAMETER (NFFTMAX=135000)! FAST FOURIER TRANSFORM GRID
        PARAMETER (NBASMAX=9000) ! BASIS SET EXPANSION IN PLANE WAVES
        PARAMETER (MBASMAX=1500) ! OPTIMAL NUMBER OF PLANE WAVES
C ************ CELL DATA PARAMETERS     **********************
        PARAMETER (NSHEMAX=60)   ! MAX.NUMBER OF POINTS IN INTERSTITIAL MESH
        PARAMETER (NMSHMAX=6000) ! MAX.NUMBER OF SURFACE GRID POINTS
        PARAMETER (NPOLMAX=12)   ! MAX.NUMBER OF POLYNOMS
C ************ FREQUENTLY USED SETTINGS **********************
        PARAMETER (NPRECISION=2) ! WITH DOUBLE PRECISION
        PARAMETER (NGGAMAX=3   ) ! WITH GGA FOR EXCHANGE-CORRELATION
        PARAMETER (NOPRMAX=48  ) ! MAX.NUMBER OF GROUP OPERATIONS
        PARAMETER (NENVMAX=30  ) ! MAX.NUMBER OF POLYHEDRON PLANES
        PARAMETER (NVCMAX=6000 ) ! MAX.NUMBER OF STRUCTURE VECTORS
        PARAMETER (NTERMAX=15  ) ! MAX.NUMBER OF BROYDEN ITERATIONS
C ************ COMBINATIONS **************************
        PARAMETER (NLENMAX=((NRADMAX+1)*4*NPAT+NPLWMAX+1)*NSPINMAX)
        PARAMETER (NWIGMAX=(LMAXT+1)*(2*LMAXT+1)*(2*LMAXT+3)/3)
        PARAMETER (LMMAXV=(LMAXV+1)**2)
        PARAMETER (LIMLMB=(LMAXB+1)**2)
        PARAMETER (LIMLMT=(LMAXT+1)**2)
        PARAMETER (LMMAXV1=(LMAXV+2)**2)
        PARAMETER (LIMLMB1=(LMAXB+2)**2)
        PARAMETER (LIMLMT1=(LMAXT+2)**2)
        PARAMETER (LMMAXV2=(LMAXV+1)*(2*LMAXV+1)*(2*LMAXV+3)/3)
        PARAMETER (LIMLMT2=(LMAXT+1)*(2*LMAXT+1)*(2*LMAXT+3)/3)
        PARAMETER (LIMLMB2=(LMAXB+1)*(2*LMAXB+1)*(2*LMAXB+3)/3)
```

8.1 Main parameters

This set of parameters is applicable for all NMT programs.

- NSPINMAX is either 1 for non-spin polarized version of the program or 2 for spin-polarized version including spin-orbit coupling (parameter *nspin* in INIFILE cannot exceed this value)
- NPAT is the maximum number of atoms allowed to consider(parameter *natom* in INIFILE cannot exceed this value) .
- NSORTMAX is maximum number of atoms of different kind (parameter *nsort* in INIFILE cannot exceed this value).
- NSYMMAX is maximum number allowed for non-zero components in the expansion of the charge density in spherical harmonics. If $(LMAXV+1)^2$ gives the total number of components, many of them can be skipped because of the symmetry of the crystall. Therefore, only non-zero components are stored inside the program. The list of terms allowed by symmetry is printed at the beginning of the OUTFILE. The maximal number of terms found for any atom must not exceed NSYMMAX. Usual choose NSYMMAX as one half of $(LMAXV+1)^2$ or so.
- LMAXV is the maximum value for the one-center spherical harmonics expansion of the charge density and the potential. (parameter *lmax-v* in INIFILE cannot exceed this value).
- LMAXB is the maximum value of l for the orbitals actually included in the basis (parameter *lmax-b* in INIFILE cannot exceed this value).
- LMAXT is the maximum value of l for the expansion of tails for the orbitals coming from different sites (parameter *lmax-t* in INIFILE cannot exceed this value). Usually LMAXT=LMAXV.
- NKAPMAX is the maximum value for the number of kappa's which can be used in a single l-channel. (parameter *nkap* in INIFILE cannot exceed this value).
- NDIMMAX is the maximum value for the dimension of the hamiltonian *without* taking into account the number of spins in case spin-orbit coupled calculation. (The H-matrix is dimensioned as NDIMMAX*NSPINMAX inside the program.) An upper limit for NDIMMAX is obviously NKAPMAX* NPAT* (LMAXB+1)**2. However, chosing different lmax-b for different atoms and/or skipping some orbitals from the basis can make NDIMMAX much smaller.
- NPANMAX is the maximum value for the number of panels (valence plus all semicore) allowed to consider.
- NRADMAX is the maximum value for the number of radial points at the logarithmic scale. If the actual value exceeds this number, the program will use NRADMAX.
- KMAX is the maximum number of **k** points allowed to generate inside the irreducible Brillouin zone.

8.2 Parameters for plane waves

This set of parameters is only effective within PLW package.

- NPLWMAX is the maximum number of plane waves allowed for the Fourier transform of the charge density and the potential in the interstitial region. If the actual number exceeds NPLWMAX, the program terminates.
- NFFTMAX is the maximum number for the FFT-grid evaluated. If the actual number evaluated as (m1+1)*(m2+1)*(m3+1) (see input FFT-grid in the INIFILE) exceeds NFFTMAX, the program terminates.
- NBASMAX is the maximum number of plane waves allowed for the Fourier transform of the LMTOs in the interstitial region. Usually it is significantly smaller than NPLWMAX.
- MBASMAX is the maximum number of plane waves allowed in evaluating interstitial-potential-matrix elements. (See the description of acceleration keys in the INIFILE) MBASMAX does not depend on the number of atoms (as NPLWMAX and NBASMAX). Usually it does not exceeds 600.

8.3 Parameters for cells

This set of parameters is effective only within the CEL package.

- NSHEMAX is the maximum number of radial points in the grid between inscribed and circumscribed sphere. Usually it does not exceed 60.
- NMSHMAX is the maximum number of polyhedron-surface grid points for evaluating the integrals over the interstitial region. The order of this parameter is 5000-10000.
- NPOLMAX is the maximum number of Chebyshev polynoms allowed for expanding the radial behavior of the interstitial quantities. Usually it is 10.

8.4 Frequently used settings

This set of parameters remains unchanged in most cases.

- NPRECISION is set to 2 (double precision), since some standard library routines are called differently depending whether single or double precision is used.
- NGGAMAX can be either 1 or 3. If NGGAMAX=1 no inclusion of GGA corrections to the LDA functional is possible. To be able to make both LDA+GGA calculations, NGGAMAX should be set to 3. This, however, affects allocated memory, since storage of first and second density gradients is required.
- NOPRMAX is always 48.
- NENVMAX is the maximum number for the planes of polyhedra allowed during their generation. Alternatively, this parameter is also used to define the maximum number of cluster sites around each atom in any tight-binding calculation.
- NVCMAX is used for the lattice summation in the structure constants. It must be at least three times larger than the input parameter *nvec* in the STRFILE.
- NTERMAX is the number of iterations after which the Broyden mixing scheme restarts. Emprically found value is 15.

8.5 Estimation of the needed core memory

The storage of the core memory is taken by several arrays. The structure constants (complex*16) are dimensioned as (LMAXT+2)**2 * NPAT * (LMAXB+1)**2 * NPAT * NKAPMAX. The charge density and the potential (complex*16) are both dimensioned as (NRADMAX+1) * NSYMMAX * NPAT * NSPINMAX. The energy bands (real*8) are stored in the array of the size NDIMMAX * NSPINMAX * KAMX * NPANMAX, The size of the hamiltonian and the overlap matrices (both complex*16) is (NDIMMAX * NSPINMAX)**2

To estimate the needed memory in bytes, multiply size of the complex array by 16, and real array by 8. A very usefull option is to link the programs to get the map-file. Under UNIX it is: **xlf *.o -o main.exe -bloadmap:map**. At the end of the map file there is a total ammount of core memory required by the program.

Disc space: no significant disc storage except for the structure constants file CONFILE. The structure constants are stored in the reduced form for all **k** points. Divide the memory taken by the structure constants approximately by four and multiply by a number of k-points for every panel to estimate the ammount of disc space required for the CONFILE.

9 ERROR MESSAGES

The description of the error messages in the program is supposed to be essentially extended in the future. Below just a few useful hints is given. Generally, two kind of errors exist in the program: warning messages (when the program does not terminate) and the error messages (when the program terminates). Normally warning messages mean that the program can either correct the problem itself or the problem is not important for the execution (like an advise to switch on scalar relativism when atomic charge exceeds 21). The error messages always mean that the program can give a wrong result if the input files will not be corrected.

9.1 Errors connected with PARAM.DAT

In all cases the program will terminate if any of the actual parameters exceeds the parameter in the PARAM.DAT. Then standard error message occurs which states which of the parameters must be increased.

9.2 Errors connected with input

Some input data can be easily checked (like the number of atoms which is read from different input files). If there is a mismatch in the input, a corresponding message is printed and execution is terminated.

9.3 Errors connected with iterative procedures

A number of iterational procedures is programmed inside the NMT package in order to find the Fermi energy or the E_ν values from a fixed set of D_ν. Limiting number of iterations and the accuracy is set in the file */nmt*/run/***setup.f**. If the number of iterations is exceeded here, the message is printed and execution is terminated.

9.4 Other errors

Some warninngs and errors are connected with the lost of accuracy in solving differential equations or in integrational procedures. Another type of errors can be due to not positevely defined overlap matrix which is most likely due to an error in the input. The overlap matrix defined with the non-overlapping MT-spheres is *always* positevely defined. When MT-spheres overlap, there is a warning message.

10 VERSIONS DIFFERENCES

This section traces differenes between current and previous versions of the program.

- *ASA1.10,CEL0.30,PLW2.01 or earlier*: not important.
- *ASA1.20,CEL0.41,PLW2.10*: Broyden mixing added.
- *ASA1.30,CEL3.50,PLW2.20*: LDA+U added.
- *ASA1.40,CEL3.62,PLW2.30*: Spin-orbit coupling added.
- *ASA1.42,CEL3.62,PLW2.32*: Relativistic solution of the semicore problem is rewritten, no references to the Liberman atomic program exist anymore.
- *ASA1.50,CEL3.70,PLW2.40*: Includes the possibilty to calculate hopping integrals for tight-binding calculations. Does not actually work well.
- *ASA1.51,CEL3.71,PLW2.41*: Marginal internal changes.
- *ASA1.52,CEL3.72,PLW2.42*: Contains LRWF key connected with the adjustment of the radial wave functions. This substitutes previously used key NOVR in the INIFILE.

Also, a bug in calculating exchange-correlation energy for IXC=4 (Vosko et.al parametrization of LSDA functional) has been found. It is not expected to influence much on the calculated results. The bug is corrected in following versions.

ASA1.60,CEL3.80,PLW2.50: Generalized gradient corrections after Perdew et.al added and tested. Two forms are avaiable here: GGA91 and GGA96.

11 USING LOCAL LIBRARY

Programs in the local libraries */nmt*/lib/* are designed for helping to construct input data for the main code and to read output files from the main code. Usually they understand input/output files only from the main code located in the same global subdirectory (e.g. */nmtplw/lib/* serves NMTPLW code, etc.) In contrast to that there exists a global library NMTLIB which contains different application programs serving any of NMT packages. See chapter USING NMTLIB LIBRARY for the detailed description.

11.1 Program CELL

Program CELL is helpful to find different sphere radii before running the main code. Ecept ASA code where the atomic sphere radii are completely determined by setting the weights for every atom (see the description of parameter *weight* in the INIFILE), the mt-sphere and circumscribed sphere must be carefully chosen before running the main program. Program is located in */nmtplw/lib/cell/* and */nmtcel/lib/cell/* directories. No specific configuring of the source files is necessary. Just compile them, link and obtain **make.exe**.

Program CELL reads INIFILE and STRFILE and prints information about the calcuated sphere radii. The basic input used for generating the spheres is again atomic weights as it has been explained above. Example of the output lines contains:

```
input mt-sphere is   2.100000     ; input as-sphere is   2.610231
cell  mt-sphere is   2.108595     ; cell  as-sphere is   2.605776
touch.mt-sphere is   2.108595     ; nrad=318,znuc= 28,zcor= 18
circums. sphere is   3.486185     ; npln= 18
```

The input mt-sphere is the one which has been read from the INIFILE. The cell mt-sphere, touch.mt-sphere and circums.sphere are those which have been generated according to the input weights. Specifically, both cell mt-sphere and the circumscribed sphere come as a result of generating the atomic polyhedra while touching spheres are just blowed up (proportionally to the weights) until they do not overlap. See also chapter OUTPUT MESSAGE FILE:OUTFILE for more information.

An important comment should be said when moving the atoms is required. Then, to avoid numerical inaccuracy, it is highly advised to use the same mt-spheres (circumscribed spheres) for all atomic geometries. To do it, one first generates STRFILEs for all needed structures, then fixes the weights in one INIFILE, computes mt-spheres (circumscribed spheres) at every geometry and then chooses the minimum values for the mt-sphere radii (maximum values for circumscribed sphere radii). This procedure allows to make sure that during atomic displacements the mt-spheres will never overlap. (For the CEL program that also means that atomic polyhedra generated for every geometry will always lie within the fixed circumscribed spheres).

11.2 Program GRID

Program GRID of the NMTPLW package is helpul to plot the charge densities, magnetization or the full potential along some line or within some plane It is located in */nmtplw/lib/grid/* directory. No special configuring of the program is required, just compile all the files, links them to get **make.exe** executable module.

The program asks for three file names: INIFILE, SCFFILE and STRFILE. Specify output charge density file as the SCFFILE if it is necessary to plot charge density or magnetization. Also specify POTFILE (see *ipot,<potfile>*) description in the chapter MAIN CONTROL FILE: INIFILE) to plot the full potential. An additional input is required to say whether the plot along the line or within the plane is required. Specify two 3D coordinates to define a desired line in the space, or specify three 3D coordinates to define a desired plane in the space. A number of divisions must be also given. The output file LINE.DAT (for line plot) and GRID.DAT (for plane plot) is formatted and contains a few columes. First one/two columns describe the line/plane grid points. The last two columns define the values of the function (charge density or potential) and the pseudofunction (pseudocharge density or pseudopotential). The values of the functions must coincide with

the values of the pseudofunction in the interstitial region. Make plots using LINE.DAT/GRID.DAT with the standard software programs.

Analogious program exists in the */nmtcel/lib/grid/* directory. Since NMTCEL uses only one-center spherical harmonic expansions no pseudocharge densities/pseudopotentials calculated. These one-center expansions often are slowly convergent and it is not advised to make good charge density plots with the CEL package.

No plotting facilities for the charge density exists in the ASA package.

11.3 Program SCF1

Program SCF1 is helpful to generate SCFFILEs for supercells. If a supercell is made of a few original cells, it first advised to make self-consistent SCFFILE for the small cell and then built input supercell SCFFILE using it. At the moment, the program SCF1 only exists for the ASA and CEL verisons of the NMT. Location: */nmtasa/lib/scf1/* and */nmtcel/lib/scf1/* directories. No special configuring is required, just compile all the programs, link them to get **make.exe** executable module.

The program asks for the SCFFILE of the original cell, then a number of atoms, number of sorts in the supercell is specified. After that, you must indicate how to copy the atoms from the old cell to the new supercell. At the end, name for the supercell SCFFILE must be given.

11.4 Program SCFM

Program SCFM is designed to construct the input charge density file (called SCFFILE) using the Mattheiss procedure by superposing free atom densities. See chapter INPUT CHARGE DENSITY FILE: SCFFILE for the complete description.

12 USING NMTLIB LIBRARY

The NMTLIB library contains a few subdirectories with the general purpose application programs. Normally, these programs understand, as input, output files from any of the NMT package. In contrast to that, inside any of the */nmt** directory there exists local library located in */nmt*/lib/*. See chapter USING LOCAL LIBRARY for the complete description.

12.1 Program ATOM

Free atom calculations are performed by Liberman's atomic program. Input files for all atoms are stored in the directory */atomdat/*; they are called "rat.<*atomname*>". An example of such input file is given below (for Fe). Most of the input parameters are self-explanatory; you do not have to care about number of (relativistic) core orbitals (see parameter **JCORE** in table below) , the NMT program will do it automatically from the known core charges of constituent atoms.

The eigenvalues given in the **rat**.<*atomname*> file are not more than the starting energies for a self-consistent free-atom calculation, so their choice is not very important. In the output, the atomic program produces files **den**.<*atomname*> with the charge density, **tab.**<*atomname*> with the table of the self-consistent eigenvalues, and **inf**.<*atomname*> with the intermediate results of the self-consistent calculation. The names of these files are explicitly given at the beginning of the **rat**.<*atomname*> file, and they can be changed. Putting 0 as the 1-st character of a file name suppresses creation of the corresponding file. Note that most of the **rat**.<*atomname*> files contain starting energies which are already self-consistent eigenvalues for atoms. They can be used as a useful reference during the work.

```
      *** FREE ATOM CALCULATION FOR FE (3D6 4S2) ***
-------------------------------------------------------
1 'den.fe'            - DENSITY FILE
1 'tab.fe'            - TABLE OUTPUT
1 'inf.fe'            - SCF-INFO FILE
10, 4                 - J,JCORE
26., 0. ,1.           - ZN, IONICITY
0.3,30                - PHI,ITERMAX
5.E-5, 1.E-3          - EPS LEVEL, EPS POT
3                     - KPOT- # OF EXCHANGE :SEE TEXT
0.                    - FEX=1:E-DEPENDENT SHAM-KOHN POTENTIAL
0.                    - ATOMIC NUMBER IF NON-POINT NUCLEUS
1.                    - INITIAL ENERGIES (1 - listed below, 0 - unknown)
-------------------------------------------------------
   ORBITAL    N    L    J     ELECTRONS     EIGENVALUE
    1S1/2    1.0  0.0  0.5       2.0     -5.1321178E+02
    2S1/2    2.0  0.0  0.5       2.0     -6.0166799E+01
    2P1/2    2.0  1.0  0.5       2.0     -5.1966747E+01
    2P3/2    2.0  1.0  1.5       4.0     -5.1051799E+01
    3S1/2    3.0  0.0  0.5       2.0     -6.9521535E+00
    3P1/2    3.0  1.0  0.5       2.0     -4.5606623E+00
    3P3/2    3.0  1.0  1.5       4.0     -4.4464996E+00
    3D3/2    3.0  2.0  1.5       4.0     -6.5538833E-01
    3D5/2    3.0  2.0  2.5       2.0     -6.4413460E-01
    4S1/2    4.0  0.0  0.5       2.0     -5.0025746E-01
```

12.2 Program CHIQ

The program CHIQ calculates imaginary/real part of the susceptibility function in the constant matrix element approximation. The corresponding formula for the real part is given by

$$Re[\chi(\mathbf{q})] = \sum_{\mathbf{k}jj'} \frac{f_{\mathbf{k}j} - f_{\mathbf{k}+\mathbf{q}j'}}{\epsilon_{\mathbf{k}j} - \epsilon_{\mathbf{k}+\mathbf{q}j'}} \tag{5}$$

The imaginary part is defined as follows:

$$Im[\chi(\mathbf{q})] = \sum_{\mathbf{k}jj'} \delta(\epsilon_F - \epsilon_{\mathbf{k}j})\delta(\epsilon_F - \epsilon_{\mathbf{k}+\mathbf{q}j'}) \tag{6}$$

No special configuring of the program is required. Just compile all the files, link them to get executable **make.exe**. The input to the program are 4 files: INIFILE, STRFILE, BNDFILE and PNTFILE. BNDFILE should contain energy bands in the irreducible part of the Brillouin zone (see the description of the input parameter *ibnd=2* in chapter describing INIFILE). PNTFILE contains a list of **q**-points for which the susceptibility is calculated. Use special program located in */nmtlib/qpnt/* directory to create PNTFILE. After the input file names are given, specify output file name, then, the following key: 1 - for the imaginary part, 2 - for the real part, 3 - for the imaginary part weighted by the velocity factor $(v_{\mathbf{k}j} - v_{\mathbf{k}+\mathbf{q}j'})^2$. The last input line should contain number of q-points to calculate (number of lines to read from PNTFILE), the Fermi energy, density of states (just to renormailize $Im[\chi(\mathbf{q})]$) and the frequency (set zero in most cases).

The output file will contain susceptibility contributions from all band transitions crossing the Fermi energy. The format of the output file coincides with the BNDFILE. Employ it if it is necessary to plot susceptibilities with help of the band programs.

12.3 Program MESH

Program MESH can expand the data from the irreducible part of the BZ to the full BZ. It is, for example, necessary when plotting the Fermi surface. Then, BNDFILE which contains energy band in the IBZ must be created (see the description of the input parameter *ibnd=2* in chapter describing INIFILE). Then use MESH program to set up energy band in the 3-dimensional cubic mesh.

No special configuring of the program is required. Just compile all the files, link them to get executable **make.exe**. The input to the program are 3 files: INIFILE, STRFILE, and BNDFILE. After that, specify three vectors and their divisions which set a cubic mesh (FFT-mesh) in the space. The output from the program is the values of the band energies at all points of this cubic mesh. Since no interpolation is assumed here, the irreducible points generated with BNDFILE must commensurate with this cubic mesh. As additional input, you must select which bands are necessary to widthdraw (e.g. only those which cross the Fermi energy). The Fermi surface can then be plotted using Data Explorer (see the description of program SURF) or other existing software.

12.4 Program MLSQ

Program MLSQ makes least-squares polynomial fit to the set of data specified in the input file. The program is useful to find different derivatives of the total energy, forces, etc. No special configuring of the program is required. Just compile the file **mlsq.f** , link it and get executable **mlsq.exe**. Sample input data file is provided.

12.5 Program PLOT

A few programs are located in this directory. Also a library of the russian GRAFOR package is located here.

- program **plobands.exe** is designed to plot energy bands along high-symmetry directions. To create an executable module, compile the files **plobands.f, calc.f**. Also compile contents of the subdirectory *GRAFOR*. Link these files to get executable **plobands.exe**.

The input to this program is BNDFILE and STRFILE. The BNDFILE must be created by any of the NMT packages with the key *ibnd*=1. The end of the STRFILE should contain a list of high-symmetry directions to plot. For example, the following control line $X-g-L, Z-R-A.$ will produce two joined pictures with the bands along $X-\Gamma$ and $\Gamma-L$ directions and sperately two joined pictures with the bands along $Z-R$ and $R-A$ directions. Small letter 'g' is understood as greek Γ. The dot at the end of the control line is understood as the end of the control line. The comma is understood as making free space between two plots. Any of the directions $X-g$, $g-L$, etc, must be calculated and stored in the BNDFILE as explained in sections describing INIFILE and STRFILE.

The output from the program has a format understandable by the SURFER/GRAPHER software. It can be directly plotted using **view.exe/plot.exe** utilities of this software. Alternatively, it can be rewritten into the postscript format using the program **plotps.exe** (see below).

- program **plotdos.exe** is designed to plot partial densities of states. To create an executable module, compile the file **pltodos.f**. Also compile contents of the subdirectory *GRAFOR*. Link these file to get executable **plotdos.exe**.

  The input to the program is given by the DOSFILE of the main NMT package. After the DOSFILE is read, you are asked which of the partial (or total) densities of states must be plotted.

  The output from the program has a format understandable by the SURFER/GRAPHER software. It can be directly plotted using **view.exe/plot.exe** utilities of this software. Alternatively, it can be rewritten into the postscript format using the program **plotps.exe** (see below).

- program **plotps.exe** is designed to reformat the output file understandable by the SURFER/GRAPHER software (plt format) to the standard postscript. No special configuring of the program is required. Just compile the file **plotps.f**, link it and get executable **plotps.exe**.

- program **fat.exe** is designed to rewrite FATFILE of the NMT package to the format understandable by the plotting facility **gnubnd.exec** of Stuttgart's TB-LMTO package. No special configuring of the program is required. Just compile the file **fat.f**, link it and get executable **fat.exe**. The input to the program is given by the FATFILE while two output files BNDS and EIGNS are created.

12.6 Program QPNT

This program is used to generate irreducible points in the Brillouin zone. No special configuring of the program is required. Just compile contents of the directory */nmtlib/qpnt/*, link object files and get executable **make.exe**. The input files to the program are INIFILE and STRFILE. The output is the number of irreducible **k**-points generated (according to the point group) and the list of points which can be stored in the output file. An additional option is provided to generate the irreducible points with its minimal lenght. This means that among a set of all $\mathbf{k}+\mathbf{G}$-points ($\mathbf{G}$ is a reciprocal lattice vector) the vector with minimal lenght will be picked out.

12.7 Program SURF

The directory SURF contains sample input files for the Data Explorer. The latter is the UNIX software which can be used to plot 3D-graphs like Fermi surfaces. Sample input files and the programs are provided.

13 ADDITIONAL INPUT: HUBFILE

WARNING: Due to a permanent development of this part of the program, some input data may differ from realization.

The LDA+U method is described in Ref. [7]. It turns out to be drastically improve the results (comparing to LDA) when doing the calculations of the strongly correlated systems. Another option avaiable here is constrained LDA calculations. See Ref. [8] for a complete description. To include LDA+U and/or LDA+C option, a special HUBFILE must be created. An example of this file for NiO system (*/nmt*/dat/nio/***nio.hub**) is given below.

```
 INPUT DATA FOR INCLUDING CORRELATED STATES
-----------------------------------------------------------------
 GENERAL SETTINGS:
 'LDA+U1.1 '                       - Scheme
 'Ry       '                       - Units eV/Ry avaiable
 'Cubic    '                       - Cubic/Spherical harmonics
 'Complex  '                       - Real/Complex input/output
 'Yes      '                       - [y/n] occupancies avaiable
 'Append   '                       - Overwrite/Append occupancies
 DESCRIPTION OF CORRELATED STATES
 2                                 - # of correlated states
 1 '3d' 0.588  0.601227  0.378773  - atom,nl,f0,f2,f4  Slater integrals
 2 '3d' 0.588  0.601227  0.378773  - atom,nl,f0,f2,f4  Slater integrals
=================================================================
 1  -  TABLE OF OCCUPATION NUMBERS FOR LDA+U :
 State 3d for Ni1; spin up/dn occupation numbers are :
      yz          zx          xy        x2-y2       3z2-1      REAL,UP,
  0.9500596   0.0000275   0.0000275  -0.0000241   0.0000139         yz
  0.0000275   0.9500596   0.0000275   0.0000241   0.0000139         zx
  0.0000275   0.0000275   0.9500596   0.0000000  -0.0000278         xy
 -0.0000241   0.0000241   0.0000000   0.9774884   0.0000000       x2-y2
  0.0000139   0.0000139  -0.0000278   0.0000000   0.9774884       3z2-1
      yz          zx          xy        x2-y2       3z2-1      IMAG,UP,
  0.0000000   0.0000000   0.0000000  -0.0000003   0.0000002         yz
  0.0000000   0.0000000   0.0000000   0.0000003   0.0000002         zx
  0.0000000   0.0000000   0.0000000   0.0000000  -0.0000004         xy
  0.0000003  -0.0000003   0.0000000   0.0000000   0.0000000       x2-y2
 -0.0000002  -0.0000002   0.0000004   0.0000000   0.0000000       3z2-1
      yz          zx          xy        x2-y2       3z2-1      REAL,DN,
  0.9440717   0.0000402   0.0000402  -0.0004752   0.0002744         yz
  0.0000402   0.9440717   0.0000402   0.0004752   0.0002744         zx
  0.0000402   0.0000402   0.9440717   0.0000000  -0.0005487         xy
 -0.0004752   0.0004752   0.0000000   0.1046295   0.0000000       x2-y2
  0.0002744   0.0002744  -0.0005487   0.0000000   0.1046295       3z2-1
      yz          zx          xy        x2-y2       3z2-1      IMAG,DN,
  0.0000000   0.0000000   0.0000000  -0.0000005   0.0000003         yz
  0.0000000   0.0000000   0.0000000   0.0000005   0.0000003         zx
  0.0000000   0.0000000   0.0000000   0.0000000  -0.0000005         xy
  0.0000005  -0.0000005   0.0000000   0.0000000   0.0000000       x2-y2
 -0.0000003  -0.0000003   0.0000005   0.0000000   0.0000000       3z2-1
 State 3d for Ni2; spin up/dn occupation numbers are :
      yz          zx          xy        x2-y2       3z2-1      REAL,UP,
```

```
 0.9440717   0.0000402   0.0000402  -0.0004752   0.0002744        yz
 0.0000402   0.9440717   0.0000402   0.0004752   0.0002744        zx
 0.0000402   0.0000402   0.9440717   0.0000000  -0.0005487        xy
-0.0004752   0.0004752   0.0000000   0.1046295   0.0000000      x2-y2
 0.0002744   0.0002744  -0.0005487   0.0000000   0.1046295      3z2-1
     yz          zx          xy        x2-y2       3z2-1      IMAG,UP,
 0.0000000   0.0000000   0.0000000  -0.0000005   0.0000003        yz
 0.0000000   0.0000000   0.0000000   0.0000005   0.0000003        zx
 0.0000000   0.0000000   0.0000000   0.0000000  -0.0000005        xy
 0.0000005  -0.0000005   0.0000000   0.0000000   0.0000000      x2-y2
-0.0000003  -0.0000003   0.0000005   0.0000000   0.0000000      3z2-1
     yz          zx          xy        x2-y2       3z2-1      REAL,DN,
 0.9500596   0.0000275   0.0000275  -0.0000241   0.0000139        yz
 0.0000275   0.9500596   0.0000275   0.0000241   0.0000139        zx
 0.0000275   0.0000275   0.9500596   0.0000000  -0.0000278        xy
-0.0000241   0.0000241   0.0000000   0.9774884   0.0000000      x2-y2
 0.0000139   0.0000139  -0.0000278   0.0000000   0.9774884      3z2-1
     yz          zx          xy        x2-y2       3z2-1      IMAG,DN,
 0.0000000   0.0000000   0.0000000  -0.0000003   0.0000002        yz
 0.0000000   0.0000000   0.0000000   0.0000003   0.0000002        zx
 0.0000000   0.0000000   0.0000000   0.0000000  -0.0000004        xy
 0.0000003  -0.0000003   0.0000000   0.0000000   0.0000000      x2-y2
-0.0000002  -0.0000002   0.0000004   0.0000000   0.0000000      3z2-1
 ===============================================================
 2  -  TABLE OF OCCUPATION NUMBERS FOR LDA   :
 State 3d for Ni1; spin up/dn occupation numbers are :
     yz          zx          xy        x2-y2       3z2-1      REAL,UP,
 0.9814361   0.0001127   0.0001127  -0.0003406   0.0001967        yz
 0.0001127   0.9814361   0.0001127   0.0003406   0.0001967        zx
 0.0001127   0.0001127   0.9814361   0.0000000  -0.0003933        xy
-0.0003406   0.0003406   0.0000000   0.9468045   0.0000000      x2-y2
 0.0001967   0.0001967  -0.0003933   0.0000000   0.9468045      3z2-1
     yz          zx          xy        x2-y2       3z2-1      IMAG,UP,
 0.0000000   0.0000000   0.0000000  -0.0000233   0.0000134        yz
 0.0000000   0.0000000   0.0000000   0.0000233   0.0000134        zx
 0.0000000   0.0000000   0.0000000   0.0000000  -0.0000269        xy
 0.0000233  -0.0000233   0.0000000   0.0000000   0.0000000      x2-y2
-0.0000134  -0.0000134   0.0000269   0.0000000   0.0000000      3z2-1
     yz          zx          xy        x2-y2       3z2-1      REAL,DN,
 0.9740245   0.0005125   0.0005125   0.0055347  -0.0031954        yz
 0.0005125   0.9740245   0.0005125  -0.0055347  -0.0031954        zx
 0.0005125   0.0005125   0.9740245   0.0000000   0.0063909        xy
 0.0055347  -0.0055347   0.0000000   0.3862361   0.0000000      x2-y2
-0.0031954  -0.0031954   0.0063909   0.0000000   0.3862361      3z2-1
     yz          zx          xy        x2-y2       3z2-1      IMAG,DN,
 0.0000000   0.0000000   0.0000000  -0.0000077   0.0000044        yz
 0.0000000   0.0000000   0.0000000   0.0000077   0.0000044        zx
 0.0000000   0.0000000   0.0000000   0.0000000  -0.0000089        xy
 0.0000077  -0.0000077   0.0000000   0.0000000   0.0000000      x2-y2
-0.0000044  -0.0000044   0.0000089   0.0000000   0.0000000      3z2-1
 State 3d for Ni2; spin up/dn occupation numbers are :
```

```
     yz          zx          xy         x2-y2       3z2-1      REAL,UP,
 0.9740251   0.0005125   0.0005125   0.0055350  -0.0031956         yz
 0.0005125   0.9740251   0.0005125  -0.0055350  -0.0031956         zx
 0.0005125   0.0005125   0.9740251   0.0000000   0.0063912         xy
 0.0055350  -0.0055350   0.0000000   0.3862469   0.0000000       x2-y2
-0.0031956  -0.0031956   0.0063912   0.0000000   0.3862469       3z2-1
     yz          zx          xy         x2-y2       3z2-1      IMAG,UP,
 0.0000000   0.0000000   0.0000000  -0.0000077   0.0000044         yz
 0.0000000   0.0000000   0.0000000   0.0000077   0.0000044         zx
 0.0000000   0.0000000   0.0000000   0.0000000  -0.0000089         xy
 0.0000077  -0.0000077   0.0000000   0.0000000   0.0000000       x2-y2
-0.0000044  -0.0000044   0.0000089   0.0000000   0.0000000       3z2-1
     yz          zx          xy         x2-y2       3z2-1      REAL,DN,
 0.9814365   0.0001126   0.0001126  -0.0003418   0.0001973         yz
 0.0001126   0.9814365   0.0001126   0.0003418   0.0001973         zx
 0.0001126   0.0001126   0.9814365   0.0000000  -0.0003947         xy
-0.0003418   0.0003418   0.0000000   0.9468114   0.0000000       x2-y2
 0.0001973   0.0001973  -0.0003947   0.0000000   0.9468114       3z2-1
     yz          zx          xy         x2-y2       3z2-1      IMAG,DN,
 0.0000000   0.0000000   0.0000000  -0.0000233   0.0000134         yz
 0.0000000   0.0000000   0.0000000   0.0000233   0.0000134         zx
 0.0000000   0.0000000   0.0000000   0.0000000  -0.0000269         xy
 0.0000233  -0.0000233   0.0000000   0.0000000   0.0000000       x2-y2
-0.0000134  -0.0000134   0.0000269   0.0000000   0.0000000       3z2-1
 ================================================================
 3  -  CONSTRAINED PART   OF THE LDA POTENTIAL :
 State 3d for Ni1; spin up/dn Hubbard potentials are :
     yz          zx          xy         x2-y2       3z2-1      REAL,UP
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         yz
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         zx
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         xy
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       x2-y2
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       3z2-1
     yz          zx          xy         x2-y2       3z2-1      IMAG,UP
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         yz
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         zx
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         xy
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       x2-y2
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       3z2-1
     yz          zx          xy         x2-y2       3z2-1      REAL,DN
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         yz
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         zx
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         xy
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       x2-y2
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       3z2-1
     yz          zx          xy         x2-y2       3z2-1      IMAG,DN
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         yz
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         zx
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         xy
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       x2-y2
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       3z2-1
```

```
 State 3d for Ni2; spin up/dn Hubbard potentials are :
      yz          zx          xy        x2-y2       3z2-1      REAL,UP
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         yz
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         zx
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         xy
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       x2-y2
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       3z2-1
      yz          zx          xy        x2-y2       3z2-1      IMAG,UP
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         yz
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         zx
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         xy
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       x2-y2
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       3z2-1
      yz          zx          xy        x2-y2       3z2-1      REAL,DN
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         yz
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         zx
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         xy
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       x2-y2
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       3z2-1
      yz          zx          xy        x2-y2       3z2-1      IMAG,DN
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         yz
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         zx
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000         xy
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       x2-y2
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000000       3z2-1
================================================================
 4  -  HUBBARD CORRECTION TO THE LDA POTENTIAL :
 State 3d for Ni1; spin up/dn Hubbard potentials are :
      yz          zx          xy        x2-y2       3z2-1      REAL,UP
 -0.2122868  -0.0000231   0.0000231   0.0000512  -0.0000295         yz
 -0.0000231  -0.2122868  -0.0000231   0.0000512   0.0000295         zx
  0.0000231  -0.0000231  -0.2122868   0.0000000   0.0000591         xy
  0.0000512   0.0000512   0.0000000  -0.2786051   0.0000000       x2-y2
 -0.0000295   0.0000295   0.0000591   0.0000000  -0.2786051       3z2-1
      yz          zx          xy        x2-y2       3z2-1      IMAG,UP
  0.0000000   0.0000000   0.0000000   0.0000002   0.0000000         yz
  0.0000000   0.0000000   0.0000000   0.0000002   0.0000000         zx
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000002         xy
 -0.0000002  -0.0000002   0.0000000   0.0000000   0.0000000       x2-y2
  0.0000000   0.0000000  -0.0000002   0.0000000   0.0000000       3z2-1
      yz          zx          xy        x2-y2       3z2-1      REAL,DN
 -0.2507736  -0.0000297   0.0000297  -0.0002219   0.0001281         yz
 -0.0000297  -0.2507736  -0.0000297  -0.0002219  -0.0001281         zx
  0.0000297  -0.0000297  -0.2507736   0.0000000  -0.0002563         xy
 -0.0002219  -0.0002219   0.0000000   0.2359186   0.0000000       x2-y2
  0.0001281  -0.0001281  -0.0002563   0.0000000   0.2359186       3z2-1
      yz          zx          xy        x2-y2       3z2-1      IMAG,DN
  0.0000000   0.0000000   0.0000000   0.0000002  -0.0000001         yz
  0.0000000   0.0000000   0.0000000   0.0000002   0.0000001         zx
  0.0000000   0.0000000   0.0000000   0.0000000   0.0000003         xy
 -0.0000002  -0.0000002   0.0000000   0.0000000   0.0000000       x2-y2
```

```
 0.0000001  -0.0000001  -0.0000003   0.0000000   0.0000000       3z2-1
State 3d for Ni2; spin up/dn Hubbard potentials are :
     yz          zx          xy        x2-y2       3z2-1      REAL,UP
-0.2507736  -0.0000297   0.0000297  -0.0002219   0.0001281         yz
-0.0000297  -0.2507736  -0.0000297  -0.0002219  -0.0001281         zx
 0.0000297  -0.0000297  -0.2507736   0.0000000  -0.0002563         xy
-0.0002219  -0.0002219   0.0000000   0.2359186   0.0000000       x2-y2
 0.0001281  -0.0001281  -0.0002563   0.0000000   0.2359186       3z2-1
     yz          zx          xy        x2-y2       3z2-1      IMAG,UP
 0.0000000   0.0000000   0.0000000   0.0000002  -0.0000001         yz
 0.0000000   0.0000000   0.0000000   0.0000002   0.0000001         zx
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000003         xy
-0.0000002  -0.0000002   0.0000000   0.0000000   0.0000000       x2-y2
 0.0000001  -0.0000001  -0.0000003   0.0000000   0.0000000       3z2-1
     yz          zx          xy        x2-y2       3z2-1      REAL,DN
-0.2122868  -0.0000231   0.0000231   0.0000512  -0.0000295         yz
-0.0000231  -0.2122868  -0.0000231   0.0000512   0.0000295         zx
 0.0000231  -0.0000231  -0.2122868   0.0000000   0.0000591         xy
 0.0000512   0.0000512   0.0000000  -0.2786051   0.0000000       x2-y2
-0.0000295   0.0000295   0.0000591   0.0000000  -0.2786051       3z2-1
     yz          zx          xy        x2-y2       3z2-1      IMAG,DN
 0.0000000   0.0000000   0.0000000   0.0000002   0.0000000         yz
 0.0000000   0.0000000   0.0000000   0.0000002   0.0000000         zx
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000002         xy
-0.0000002  -0.0000002   0.0000000   0.0000000   0.0000000       x2-y2
 0.0000000   0.0000000  -0.0000002   0.0000000   0.0000000       3z2-1
```

13.1 General Settings

No case sensitivity is assumed in the input parameters described below.

```
'LDA+U1.1 '                    - Scheme
'Ry       '                    - Units eV/Ry avaiable
'Cubic    '                    - Cubic/Spherical harmonics
'Complex  '                    - Real/Complex input/output
'Yes      '                    - [y/n] occupancies avaiable
'Append   '                    - Overwrite/Append occupancies
```

- *Scheme*: A number of different formulae have been programmed for the LDA+U technique. For a complete description see also file */nmt*/run/***hubpot.f**.
  - 'LDA ': standard LDA. Can be used to withdraw LDA occupation numbers matrix.
  - 'LDA+U1.1': standard LDA+U. Starting occupancies must be listed (see below).
  - 'LDA+U1.2': LDA+U with another double counting. Together with the starting occupancies LDA occupation numbers must be given (see below).
  - 'LDA+U1.3': LDA+U with the average occupanicies.
  - 'LDA+U1.4': LDA+U in spherically averaged form.
  - 'LDA+C ': constrained LDA calculations. A constrained part of the potential must be specified (see below).

– 'LDA+CU1.1': combination of constrained LDA with the LDA+U techniqie (scheme 1.1, 1.2, 1.3 or 1.4)

- *Units*: Electronvolts or Ry input units for the energies in this file are avaiable. Specify either 'eV' or 'Ry'.
- *Harmonics*: The input can be given either in 'spherical' or in 'cubic' harmonics representation.
- *Input/Output* can be either 'real' or 'complex'.
- *Occupancies* in this file are either listed below ('yes' key) or absent ('no' key).
- *Occupancies* in this file will be either overwritten during the execution ('overwrite' option) or appended ('append' option) to the end of the file. The latter is recommended for use.

13.2 Description of Correlated States

```
2                                  - # of correlated states
1 '3d' 0.588  0.601227  0.378773  - atom,nl,f0,f2,f4  Slater integrals
2 '3d' 0.588  0.601227  0.378773  - atom,nl,f0,f2,f4  Slater integrals
```

- # of states which are considered as correlated must be given.
- for each of the correlated state (selected by atom number, main quantum number, orbital quantum number) Slater integrals must be given. For d-electrons the knowledge of on-site Coulomb U and exchange integral J defines set of Slater integrals as follows: $U = F^{(0)}$,$J = (F^{(2)} + F^{(4)})/14$, and $F^{(2)}/F^{(4)} = 0.625$.

13.3 Table of Occupation Numbers

At the end of the HUBFILE matrix of the occupation numbers and/or correction to the LDA potential for each of the correlated state must be given.

A specific input is selected by the first key:

- 1 stands for the input occupation matrix for each correlated state.
- 2 stands for the occupation matrix as calculated within LDA. This part of the input can be skipped unless Scheme='LDA+U1.2' is switched on.
- 3 stands for the correction to the LDA potential in constrained LDA calculations. This part of the input can be skipped unless Scheme='LDA+C' is switched on.
- 4 stands for the correction to the LDA potential as given in LDA+U calculation. These data are not used as input and for information only. Therefore, they can be ommited while constructing HUBFILE.

14 ADDITIONAL INPUT: HOPFILE

WARNING: Due to a permanent development of this part of the program, some input data may differ from realization.

This application is designed to calculate hopping mattrix elements (for tight-binding description) by the NMT program. The development is not yet finished although some possibility for rough evaluation is provided.

To be able to calculate hopping integrals, a special HOPFILE (located in */nmt*/dat/nio/***nio.hop**) must be created. Also localization sphere radii must be set up in the INIFILE (See the description of the input parameter *rloc* in the INIFILE)

Below is an example of HOPFILE:

```
 INPUT DATA FOR TIGHT-BINDING CALCULATION
-----------------------------------------------------------------
GENERAL SETTINGS:
 'Bands    '                      - Scheme
 'Ry       '                      - Units eV/Ry avaiable
 'Cubic    '                      - Cubic/Spherical harmonics
 'Real     '                      - Real/Complex input/output
 'Yes      '                      - Yes/No hoppings avaiable
 'Append   '                      - Overwrite/Append new hoppings
DESCRIPTION OF HOPPING INTEGRALS:
 3 From :         to site:                    Energy:    Overlap:
 Ni1#1 3d{x2-y2}  Ni1@[0,  0,  0  ] 3d{x2-y2}  -0.301       1.0
 O  #3 2p{z}      O  @[0,  0,  0  ] 2p{z}      -0.140       1.0
 Ni1#1 3d{x2-y2}  O  @[1/2,1/2,1/2] 2p{z}      +0.162       0.0
```

14.1 General Settings

No case sensitivity is assumed in the input parameters described below.

- *Scheme*: Only Scheme='Bands' is avaiable at the moment.
- *Units*: Electronvolts or Ry input units for the energies in this file are avaiable. Specify either 'eV' or 'Ry'.
- *Harmonics*: The input can be given either in 'spherical' or in 'cubic' harmonics representation.
- *Input/Output* can be either 'real' or 'complex'.
- *Hoppings* in this file either listed below ('yes' key) or absent ('no' key).
- *Hoppings* in this file will be either overwritten during the execution ('overwrite' option) or appended ('append' option) at the end of the HOPFILE.

14.2 Description of Hopping Integrals

In this part of the input you must select which of the hopping matrix elements to calculate. Hopping element is defined between two (or the same) orbitals. In the first line, total number of hopping elements which is supposed to calculate must be given.

- To select orbital "from" specify a sort title, then atom number as listed in STRFILE, then main quantum number, orbital quantum number (s for l=0, p for l=1, etc), then magnetic quantum number in brackets {}. Values of $m = -3, -2, -1, 0, +1, +2, +3$ are readable for spherical harmonics. For cubic harmonics use *x,y,z* when l=1, or *yz, zx, xy, x2-y2, 3z2-1* when l=2, or *x(5x2-3), y(5y2-3), z(5z2-3), y(x2-z2), z(x2-y2), x(y2-z2), xyz* when l=3. See also file */nmt*/run/***cubharm.f** for definition of cubic harmonics.

- To select orbital "to site" use the same notations as above except for the position of atom in the unit cell. Specify a vector which connect site "From" with site "to". The format to set this vector is given in the example above. Any of expressions understandable by *calculator* can be used to set coordinates. For the description of *calculator* see section STRUCTURE CONTROL FILE.
- *Energy* is the hopping matrix element. It is not input parameter to the main program, but it will be overwritten (or appended) during the calculation.
- *Overlap* is the overlap matrix element between these two orbitals, for orthogonal representation it is either 1 for the same orbital and 0 otherwise. It is not input parapeter.

The HOPFILE can be used as input file to the package FTBHUB which is not automatically distributed with the NMT package, but can be obtained upon request. The FTBHUB program performs non-orthogonal tight-binding calculations with the hopping integrals read from HOPFILE and the Coulomb on-site interactions read from HUBFILE.

15 ADDITIONAL INPUT: KOVFILE

If it is necessary to built your own rotational system, the file describing rotational operations (KOVFILE) must be created. Two examples of this file are given below for cubic and hexagonal groups. Note that both cubic and hexagonal symmteres are understandable automatically if the symmetry code is set to either 'C' or 'H' (see the description of the symmetry code in the chapter describing STRFILE) However the same effect can be reached if the symmetry code is set to 'A' and one of the KOVFILEs shown below is created. You can make your own KOVFILE using these examples.

KOVFILE for describing the *cubic* rotational system is listed below. All the expressions must be recognizable by the CALC facility (see the description of the calculator in the chapter describing STRFILE) Cubic rotations are given after Kovalev. Note the difference between the present and Kovalev numbering which is given by a second column. Notations are the following:

- U - equivalent operation
- A - arbitrary rotation along (X,Y,Z) (except Z-axe)
- Z - rotation along Z
- I - inversion
- C - combination

```
48    - total # of operations in the Cubic group
U     - operation: equivalent
1
A     - operation: rotations along arbitrary axe
20    - element #
pi/2
+1, 0, 0
A     - operation
2
pi
+1, 0, 0
A     - operation
19
3*pi/2
+1, 0, 0
A     - operation
22
pi/2
0,+1, 0
A     - operation
3
pi
0,+1, 0
A     - operation
24
3*pi/2
0,+1, 0
Z     - operation, rotations along Z-axe
15
pi/2
Z     - operation
```

```
4
pi
Z     - operation
14
3*pi/2
A     - operation
5
2*pi/3
+1,+1,+1
A     - operation
9
4*pi/3
+1,+1,+1
A     - operation
10
2*pi/3
+1,-1,+1
A     - operation
8
4*pi/3
+1,-1,+1
A     - operation
6
2*pi/3
-1,-1,+1
A     - operation
11
4*pi/3
-1,-1,+1
A     - operation
12
2*pi/3
-1,+1,+1
A     - operation
7
4*pi/3
-1,+1,+1
A     - operation
16
pi
+1,+1, 0
A     - operation
13
pi
+1,-1, 0
A     - operation
18
pi
 0,+1,+1
A     - operation
17
```

```
pi
 0,-1,+1
A     - operation
23
pi
+1, 0,+1
A     - operation
21
pi
-1, 0,+1
I     - operation: inversion
25
C     - operation: combinations
26
25 2
C     - operation
27
25 3
C     - operation
28
25 4
C     - operation
29
25 5
C     - operation
30
25 6
C     - operation
31
25 7
C     - operation
32
25 8
C     - operation
33
25 9
C     - operation
34
25 10
C     - operation
35
25 11
C     - operation
36
25 12
C     - operation
37
25 13
C     - operation
38
25 14
```

```
C     - operation
39
25 15
C     - operation
40
25 16
C     - operation
41
25 17
C     - operation
42
25 18
C     - operation
43
25 19
C     - operation
44
25 20
C     - operation
45
25 21
C     - operation
46
25 22
C     - operation
47
25 23
C     - operation
48
25 24
```

KOVFILE for describing the *hexagonal* rotational system is listed below. All the expressions must be recognizable by the CALC facility (see the description of the calculator in the chapter describing STRFILE). Hexagonal rotations are given after Kovalev. Note the difference between the present and Kovalev numbering which is given by a second column. Notations are the following:

- U - equivalent operation
- A - arbitrary rotation along (X,Y,Z) (except Z-axe)
- Z - rotation along Z
- I - inversion
- C - combination

```
24    - total # of elements in the Hexagonal group
U     - operation: equivalent
1
Z     - operation: rotations along Z
2     - element #
pi/3
Z     - operation
3
```

```
2*pi/3
Z     - operation
4
pi
Z     - operation
5
4*pi/3
Z     - operation
6
5*pi/3
A     - operation
10
pi
+1,0,0
A     - operation: rotations along arbitrary axe
9
pi
+sqrt(3)/2,+1/2,0
A     - operation
8
pi
+1/2,+sqrt(3)/2,0
A     - operation
7
pi
0,+1,0
A     - operation
12
pi
-1/2,+sqrt(3)/2,0
A     - operation
11
pi
-sqrt(3)/2,+1/2,0
I     - operation: inversion
13
C     - operation: combinations
14
13 2
C     - operation
15
13 3
C     - operation
16
13 4
C     - operation
17
13 5
C     - operation
18
13 6
```

```
C     - operation
19
13 7
C     - operation
20
13 8
C     - operation
21
13 9
C     - operation
22
13 10
C     - operation
23
13 11
C     - operation
24
13 12
```

16 Acknowledgents

I greatly acknowledge Dr. Andrej Postnikov who has initiated writting of this manual. Part of the developments has been done in collaboration with my brother Dr. Dmitrij Savrasov. Special thanks to Prof. Ole Andersen and Dr. Ove Jepsen who are my LMTO teachers.

17 COPYRIGHT

These programs are a free software for scientific and/or educational purposes.

It is not allowed to redistribute them without prior written consent of the Copyright owners.

It is illegal to commercially distribute these programs as a whole or incorporate any part of it into a commercial product.

References

[1] O. K. Andersen, Phys. Rev. B **13**, 3050 (1975).

[2] P. Hohenberg and W. Kohn, Phys. Rev. **136**, B864 (1964).

[3] W. Kohn and L. J. Sham, Phys. Rev. **140**, A1133 (1965).

[4] For a review, see, also , *Theory of the Inhomogeneous Electron Gas*, edited by S. Lundqvist and S. H. March (Plenum, New York, 1983).

[5] S. Y. Savrasov and D. Y. Savrasov, Phys. Rev. B. **46**, 12181 (1992).

[6] S. Y. Savrasov, Phys. Rev. B **54**, 16470 (1996).

[7] For a review and complete set of references see, e.g., V. Anisimov, F. Aryasetiawan, and A. I. Lichtenstein, J. Phys.: Condens. Matter **9**, 767 (1997).

[8] M. S. Hybertsen, M. Schlüter, and N. E. Christensen, Phys. Rev. B **39**, 9028 (1989).